An Oracle White Paper

September 2014

Extending Oracle E-Business Suite Release 12.1 and above using Oracle Application Express

Revision 2

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Executive Overview

This paper outlines how to extend Oracle E-Business Suite functionality utilizing Oracle Application Express. This paper is a collaboration between both product teams to ensure that any extensions developed following the strategy outlined in this paper are fully supported. Recommended architecture and security considerations are discussed in detail.

Introduction

Oracle E-Business Suite delivers a wide range of functionality to handle core areas of your business processing needs. However, there are situations where you want to extend your information systems beyond the range of Oracle E-Business Suite. Many times these necessary extensions are meant to handle unique industry conventions, specific customer requirements, or perhaps to offer some other competitive edge. Sometimes these change requests are simple enough, but other times more extensive customizations are needed. In these scenarios, Oracle Application Express, also known as Oracle APEX, provides an easy way to create supplemental applications that are easily integrated with your Oracle E-Business Suite and its data.

Oracle Application Express is a rapid web application development tool for the Oracle database. Oracle Application Express combines the qualities of a personal database (productivity, ease of use, and flexibility) with the qualities of an enterprise database (security, integrity, performance, scalability, availability, and built for the web). The browser based interface, declarative programming framework, and simple wizards make Oracle Application Express easy to learn and enable you to quickly build robust applications.

Oracle Application Express offers you a quick and highly productive way to extend your Oracle E-Business Suite environment with almost no impact to your existing implementation. By moving custom behaviors out of Oracle E-Business Suite and into Oracle Application Express, you can make Oracle E-Business Suite patching and upgrading much simpler.

You should be able to easily combine your Oracle Application Express applications with your Oracle E-Business Suite installation using the architecture and techniques described in this white paper.

Prerequisites

The prerequisites for the solution given in this paper are:

- Oracle E-Business Suite 12.1.3 or above
  - This paper includes specifics instructions that apply to both Oracle E-Business Suite Release 12.1 and Release 12.2
  - For Oracle E-Business Suite Release 12.1, patch 9659153:R12.FND.B is required. {Go to My Oracle Support Patches & Updates: https://updates.oracle.com/Orion/PatchSearch/process_form?bug=12316083}
- Fully Licensed Oracle Database[1], Release 10.2.0.3 or above
- Oracle Application Express[2] 4.2 or above
- Oracle Rest Data Services[3] 2.0 or above
- Oracle WebLogic Server[4] 12c or above
  - Oracle Glassfish Server and Apache Tomcat can also be utilized. {See Oracle Application Express Architecture for more details}

The examples in this document use Oracle Application Express 4.2.6. The exact steps and screen shots will be different for later versions of Oracle Application Express.

[1] The limited-use Oracle Database license agreement included with an Oracle E-Business Suite license does not allow for the creation of an additional schema as outlined in this paper below. Please refer to the following license agreement: http://www.oracle.com/us/corporate/pricing/application-licensing-table-070571.pdf. Therefore, to be fully compliant, full-use Oracle Database Enterprise Edition and Oracle Internet Application Server Enterprise Edition licenses are required.

[2] For customers who used earlier revisions of this paper to integrate Oracle E-Business Suite Release 12.1 and Oracle Application Express, Oracle Application Express 4.0 and above is still supported when integrating with Oracle E-Business Suite Release 12.1.

[3] For customers who used earlier revisions of this paper to integrate Oracle E-Business Suite Release 12.1 and Oracle Application Express, Oracle APEX Listener 1.1 and above is still supported when integrating with Oracle E-Business Suite Release 12.1.

[4] A separate Oracle WebLogic Server license is required, other than the Oracle WebLogic Suite for Oracle Applications, utilized for Oracle E-Business Suite Release 12.2.

Concept Overview

Oracle Application Express offers you the ability to create reports, charts, and calendars as well as pages to provide for data review and manipulation. To allow you to create Oracle Application Express applications that reference Oracle E-Business Suite data, the desired data is exposed through database views in the APPS schema. 'SELECT' access to those views is then granted to a new schema (for this document, APEX_EBS_EXTENSION). Because direct ‘INSERT’, ’UPDATE’, and ’DELETE’ operations on Oracle E-Business Suite data is unsupported, applications that manipulate Oracle E-Business Suite data will use the public, documented and supported, Oracle E-Business Suite APIs. References to these APIs can either be coded into the Oracle Application Express applications or you can define 'INSTEAD OF' triggers against the new database views which call the appropriate APIs.

The extensions built in Oracle Application Express will appear seamless to your end users because you can use the same authentication (who can login) and authorization (who can see what) within your Oracle Application Express applications that are used within your Oracle E-Business Suite installation. Your new applications can either be stand-alone or fully integrated. Stand-alone applications are accessed directly, but use the same login credentials used for Oracle E-Business Suite access. Fully integrated applications are registered within the Oracle E-Business Suite so they are available from within the Oracle E-Business Suite menus. Oracle Application Express contains a pre-configured Authentication Scheme for Oracle Single Sign-on (OSSO) and also HTTP Header Variable (Oracle Application Express Release 4.1 and above) for use with Oracle Access Manager (OAM).

If your Oracle E-Business Suite instance uses OSSO or OAM for authentication, setting up authentication will be very simple. If you are using custom authentication for Oracle E-Business Suite, you can still achieve seamless integration but you will need to code a function within the APPS schema that validates the user, and then add an authentication scheme to your Oracle Application Express applications that calls that new function.

This document discusses the Oracle Application Express architecture and provides detailed instructions for creating all the components necessary to extend Oracle E-Business Suite. In support of the instructions, scripts and sample packaged applications are available for download to ensure that any developer can quickly replicate the examples.

Oracle Application Express Architecture

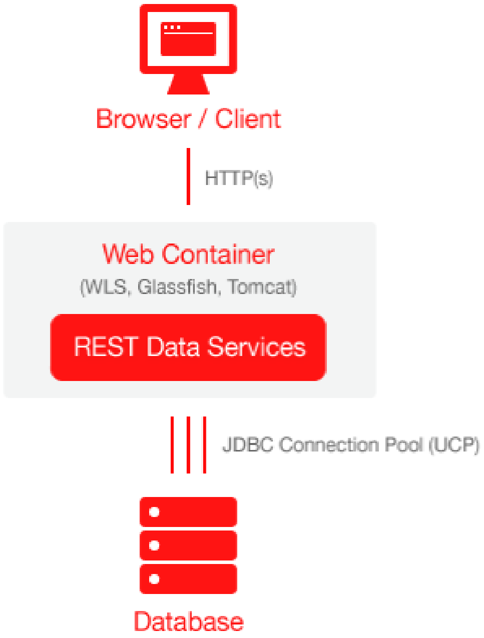


Figure 1. Oracle Application Express Architecture

Oracle Application Express resides completely within the Oracle Database in its own schema and can be installed on any version of the Oracle Database from 10gR2 and above. Runtime, development and deployment require no client software as access is 100% browser based via a Web listener communicating with the Oracle Database. The application definitions are stored as meta-data within the Oracle Application Express schema which is accessed to perform page rendering and processing.

There are currently three Web listeners that can be utilized to run Oracle Application Express – Oracle REST Data Services (formerly the APEX Listener), Oracle HTTP Server with mod_plsql, or the Embedded PL/SQL Gateway (EPG) available with Oracle Database 11gR1 and above. The Oracle REST Data Services is a JAVA EE based solution that is certified with Oracle WebLogic Server, Oracle Glassfish Server and Apache Tomcat. Oracle REST Data Services is the preferred Web listener for running Oracle Application Express.

Note: It is not recommended to use the Embedded PL/SQL Gateway within the E-Business Suite Release environment as it will add additional load on the Database Server.

Recommended Mid-Tier Deployment Architecture

The recommended configuration is to have two separate application servers, by installing the Oracle REST Data Services within Oracle WebLogic Server, on a separate application server to the one configured for Oracle E-Business Suite. {See Figure 2 and Figure 3}. Oracle recommends installing a separate application server so that each application server can be managed independently.

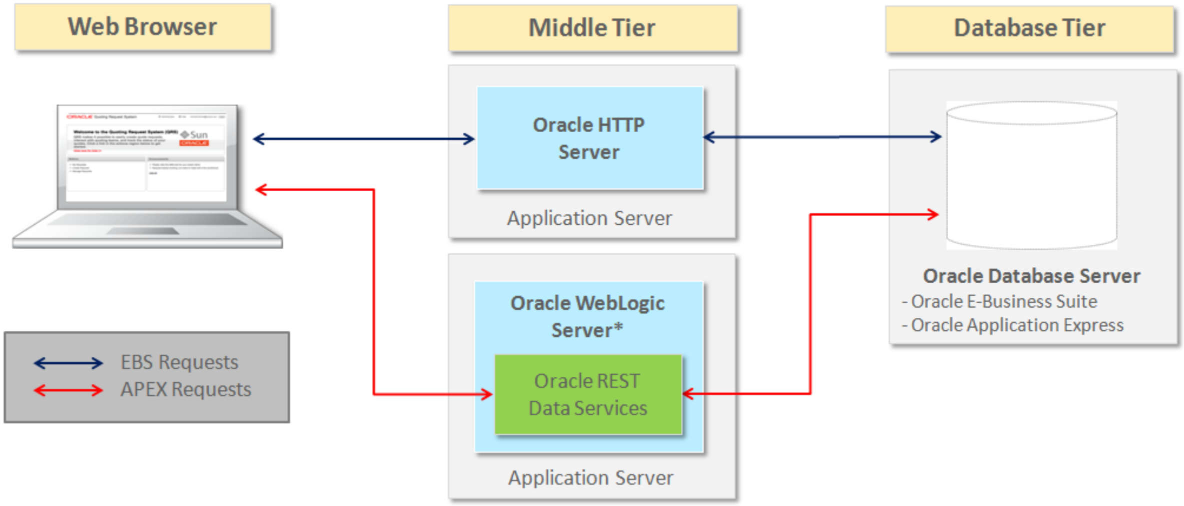


Figure 2. Recommended Middle Tier Configuration - <u>Release 12.1</u>

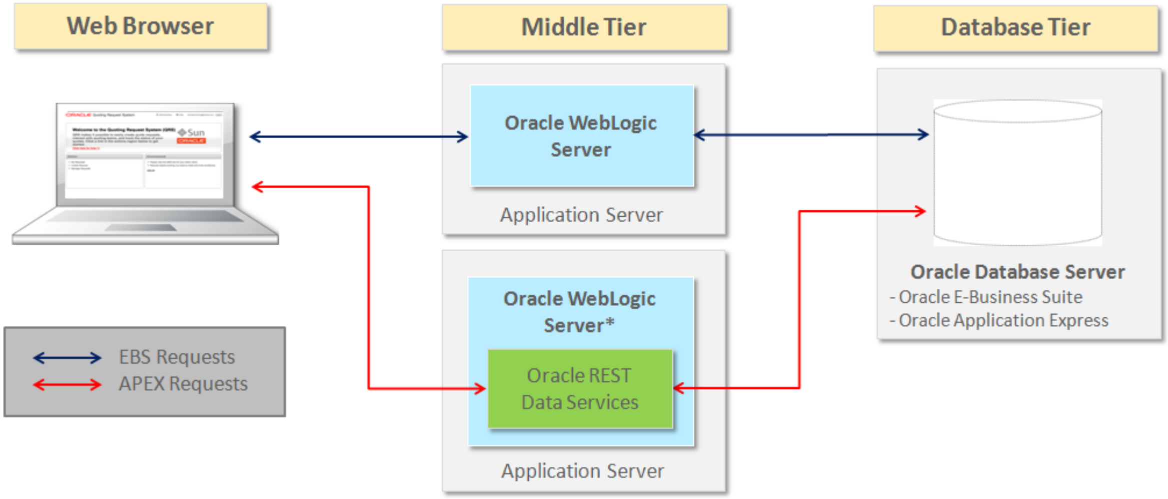


Figure 3. Recommended Middle Tier Configuration - <u>Release 12.2</u>

Note: You can also install Oracle REST Data Services into Oracle Glassfish Server or Apache Tomcat, instead of Oracle WebLogic Server, and still be fully supported and in compliance.

Alternate Mid-Tier Deployment Architecture for E-Business Suite Release 12.1

For E-Business Suite Release 12.1 you can also install the Oracle WebLogic Server on the same Application Server utilized by Oracle E-Business Suite {See Figure 4}. However, it is not certified to install Oracle REST Data Services into the Oracle WebLogic Server(s) utilized by Oracle E-Business Suite Release 12.2.

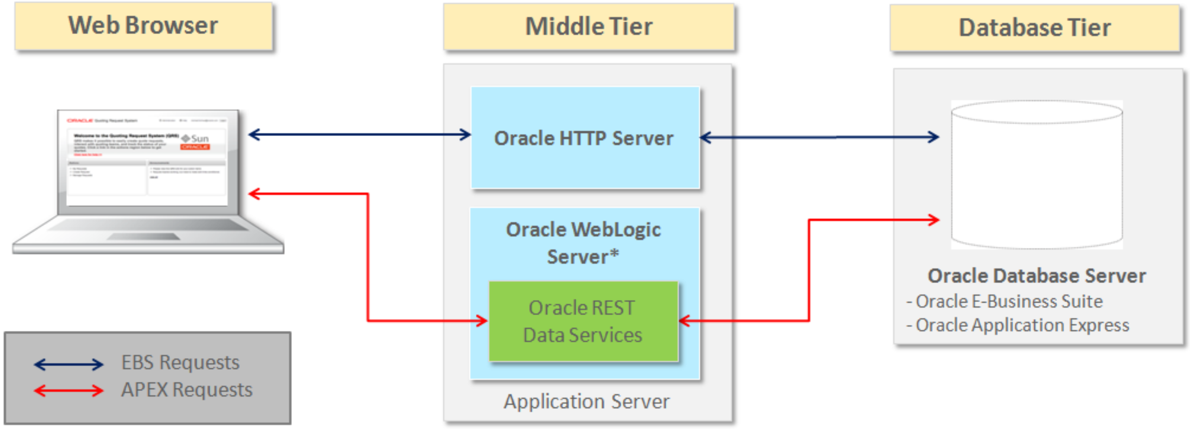


Figure 4. Alternate Middle Tier Configuration - Release 12.1 ONLY

Unsupported Mid-Tier Deployment Architecture

Oracle E-Business Suite Release 12.1 disables the mod_plsql gateway within the Oracle HTTP Server configured for Oracle E-Business Suite. Enabling mod_plsql for use with Application Express is not recommended and will result in your Oracle E-Business Suite configuration not being certified or supported.

Recommended Database Architecture

Oracle recommends configuring Oracle Application Express within the Oracle Database server where Oracle E-Business Suite is installed {See Figure 5}. A separate, minimally privileged, schema (APEX_EBS_EXTENSION in this document) must be defined as the basis for building Oracle Application Express applications. Defining an Oracle Application Express workspace directly against the owning Oracle E-Business Suite schema (APPS) is not supported.

This recommendation is primarily based on maximizing the performance of the queries and data operations within Oracle Application Express applications. The Oracle Database is exceedingly efficient at retrieving data from the same database instance, even if the data is being sourced from a different database schema.

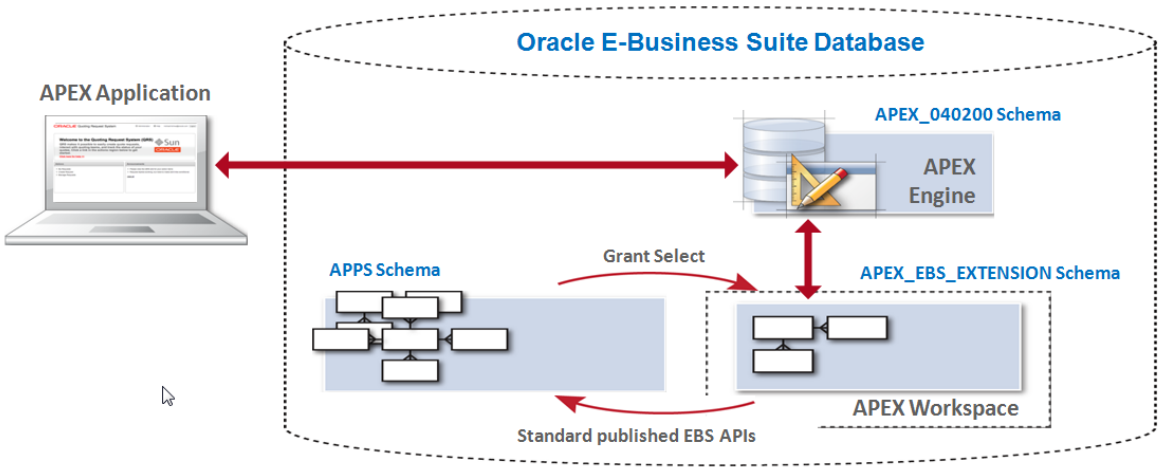

Oracle E-Business Suite Database
APEX Application
APEX_040200 Schema
APEX Engine
Grant Select
APPS Schema
APEX_EBS_EXTENSION Schema
APEX Workspace
Standard published EBS APIs

Figure 5. Recommended Database Configuration

Alternative Database Architecture using Database Links

You are permitted to configure Oracle Application Express on a separate Oracle Database server from the Oracle Database server where Oracle E-Business Suite is installed {See Figure 6}. A private database link from any external Oracle Database server to the Oracle E-Business Suite Database server is recommended. This database link should be made to a separate, minimally-privileged schema, from the owning Oracle E-Business Suite schema (APPS). It is imperative that you do not define database links directly to the owning Oracle E-Business Suite schema (APPS), as this is a breach of security protocol.

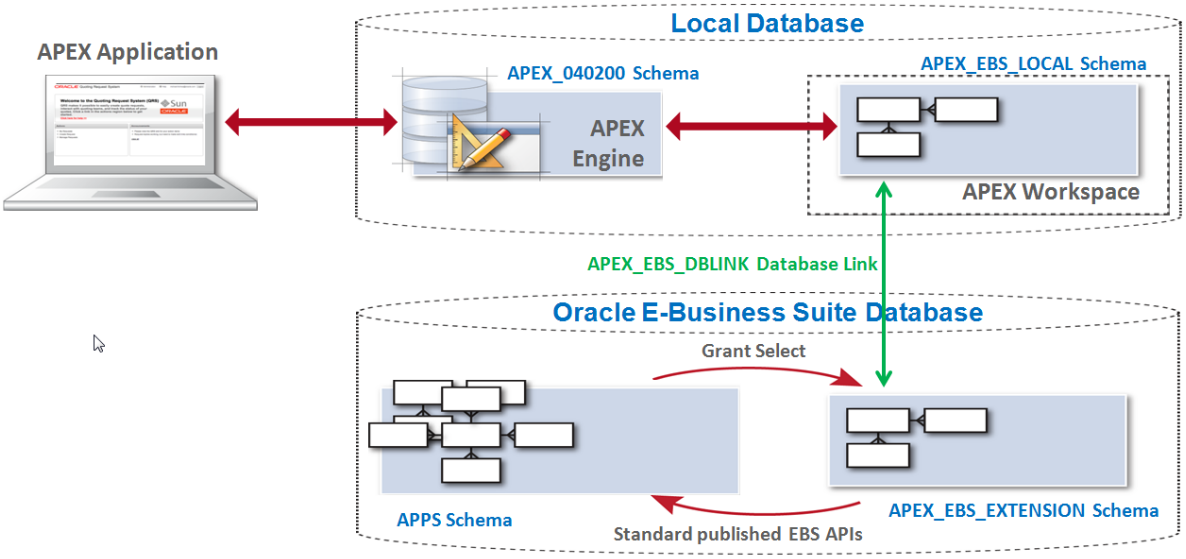


Figure 6. Alternate Database Configuration

For example, if you wish to enhance existing Oracle Application Express applications, that are running on another Oracle Database, to incorporate Oracle E-Business Suite information, then you can utilize database links to obtain the required Oracle E-Business Suite data.

Upgrading to Oracle E-Business Suite Release 12.2 when Oracle Application Express Already Integrated

If you have previously integrated Oracle Application Express with Oracle E-Business Suite Release 12.1 or earlier, and are upgrading to Oracle E-Business Suite Release 12.2, then you need to plan your upgrade to take into consideration the integration of these products. Specifically Oracle recommends the following tasks:

- Upgrade to E-Business Suite Release 12.2 following the My Oracle Support Knowledge Document 1531121.1
  - Register the schema(s), assigned to Application Express workspaces, within E-Business Suite Release 12.2. For further details, review Chapter 4 “Oracle E-Business Suite Online Patching” in the Oracle E-Business Suite R12.2 Upgrade Guide
- Upgrade to the latest version of Oracle Application Express (currently 4.2.6) available on the Oracle Technology Network (http://otn.oracle.com/apex).
- Upgrade to the latest version of Oracle REST Data Services (currently 2.0.9)

Before running the Oracle E-Business Suite Online Enablement process, it is critical that the Application Express schema(s) conform to Online Patching Standards as defined in My Oracle Support Knowledge Document 1531121.1. Otherwise, your Application Express extensions will stop working. With Oracle E-Business Suite Release 12.2 the APPS schema will be Edition-Based Redefintion (EBR) enabled to facilitate E-Business Suite Online Patching. Attempting to access editioned objects from a non-editioned schema will cause Oracle errors. Oracle E-Business Suite Online Patching will now correctly edition the Application Express workspace schema(s) so that your Application Express extensions continue to function without interruption during the online upgrade process.

> Note: If you attempt to enable edition-based redefinition on the Application Express Engine schema (For Application Express Release 4.2 that schema is APEX_040200) you will invalidate Application Express, making it inoperable.

Oracle recommends upgrading from any previous releases of Oracle Application Express and Oracle REST Data Services to the latest available releases of each. This can be performed by following the appropriate installation guides for each product. These updates can be performed as part of the 12.2 Post Upgrade steps, before the system is released to end users. Future updates to these components could also be scheduled as part of an E-

Business Suite Online Patching Cycle, after performing a cutover phase that does not bring up the middle tier processes (adop mtrestart=no). For further details on Online Patching and the adop utility, review the Oracle E-Business Suite R12.2 Maintenance Guide, “The adop utility” section.

If you were previously running Oracle REST Data Services (formerly known as the APEX Listener) on Oracle Glassfish, then you may continue to utilize that application server. However, you should review the support policy related to Oracle Glassfish as this has changed since the previous version of this white paper was released. Oracle recommends that you install Oracle REST Data Services into Oracle WebLogic Server, but if you choose to utilize Oracle Glassfish or Apache Tomcat you will still be in compliance.

Now that you have upgraded your environment you should review the Upgrading your Application Express Extensions section below, to understand how to deploy enhancements to your Application Express applications, especially when they involve modification to existing database objects.

Installing Oracle Application Express

For a fresh install, when you have not previously built Application Express extensions within your Oracle E-Business Suite environment, you need to configure Oracle Application Express within the Oracle Database. Since Oracle Database 11gR1, Oracle Application Express is part of the standard seed database installation. As such the Application Express engine is probably already installed but not configured within your environment. However, Oracle very highly recommends downloading the latest version of Oracle Application Express (currently APEX 4.2.6) from the Oracle Technology Network (http://otn.oracle.com/apex) and following the Installation Guide – 3.3 Downloading from OTN and Configuring Oracle Application Express Listener (http://docs.oracle.com/cd/E37097_01/doc/install.42/e35123/otn_install.htm#BABJJAGF). {Application Express Listener was renamed to Oracle REST Data Services after the installation guide for Application Express 4.2 was delivered}.

You will also need to download the latest version of Oracle REST Data Services (currently 2.0.9) from Oracle Technology Network (http://www.oracle.com/technetwork/developer-tools/rest-data-services/overview/index.html) and follow the installation instructions for the Oracle WebLogic Server. To maintain the same context path for Application Express, `/apex`, you must rename the `ords.war` file to `apex.war` before deployment within Oracle WebLogic Server.

After completing the installation steps for both Oracle Application Express (including enabling Network Services in Oracle Database 11g) and Oracle REST Data Services, you need to create a workspace for your development and then create applications within that workspace. For your production instance it is highly recommended to run Oracle Application Express in runtime-only mode to further harden security [See: About the Oracle Application Express Runtime Environment http://docs.oracle.com/cd/E37097_01/doc/install.42/e35123/overview.htm#CJAFIGFG].

Oracle Application Express Schemas and Workspaces

Workspaces are logical containers within Oracle Application Express that provide functional security. Workspaces can be associated with one or more database schemas. Each schema associated with an Oracle Application Express workspace can be used to parse SQL and PL/SQL requests.

Oracle E-Business Suite uses a schema called APPS. For better security, Oracle strongly discourages associating the APPS schema with an Oracle Application Express workspace. Associating the APPS schema allows Oracle Application Express applications full access to all of the underlying Oracle E-Business Suite tables, which is a security risk. Instead, create at least one separate schema in the Oracle E-Business Suite database for developing your Oracle Application Express applications.

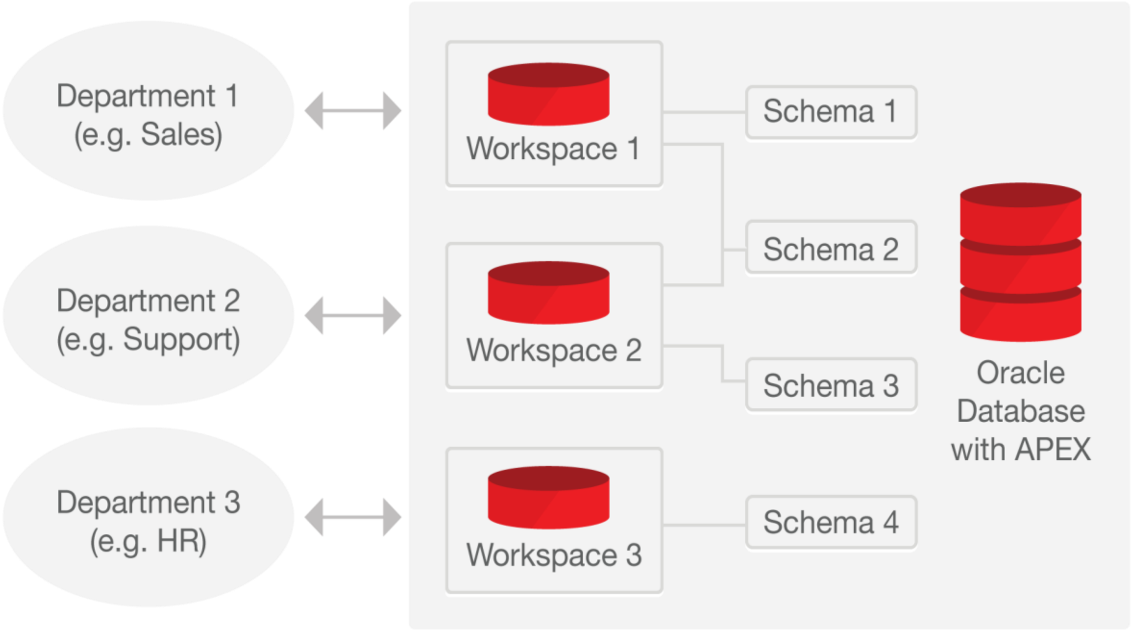


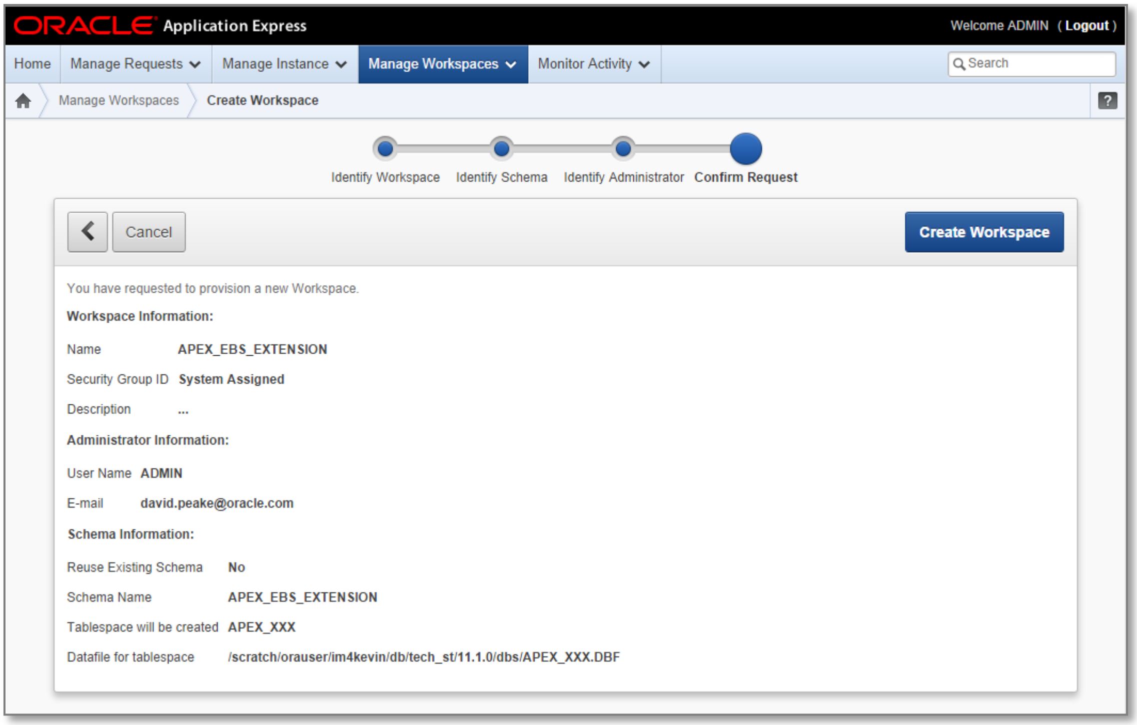
Figure 7. Example of Oracle Application Express Workspaces

Defining the Oracle Application Express Workspace and Schema

Based on the recommended database architecture, you should create a workspace called APEX_EBS_EXTENSION and a schema called APEX_EBS_EXTENSION on the database server where Oracle E-Business Suite is installed. However, if you plan to develop Application Express on a different server then skip this section and follow the steps outlined in Configuring Oracle Application Express in a Different Database.

Create a workspace called APEX_EBS_EXTENSION and a schema called APEX_EBS_EXTENSION by performing the following steps:

1. Log into Oracle Application Express Administration (**http://<EBS_Hostname>:8080/apex/apex_admin**)
2. Enter *Username* **ADMIN**, *Password* (defined during installation by executing @apxchpwd), click Login
3. Navigate to Manage Workspaces > Create Workspace
4. Enter *Workspace Name* **APEX_EBS_EXTENSION**, click *Next >*
5. Enter *Schema Name* **APEX_EBS_EXTENSION**, and *Password* ***<<apex_ebs_extension>>*** , click *Next >*
6. Enter *Administrator Username* **ADMIN**, *Administrator Password* <<***APEX_Password>>***, and your email address for *Email*, click *Next >*
7. Click Create Workspace, click *Done*



You have requested to provision a new Workspace.

Workspace Information:

Name	APEX_EBS_EXTENSION
Security Group ID	System Assigned
Description	...

Administrator Information:

User Name	ADMIN
E-mail	david.peake@oracle.com

Schema Information:

Reuse Existing Schema	No
Schema Name	APEX_EBS_EXTENSION
Tablespace will be created	APEX_XXX
Datafile for tablespace	/scratch/orauser/im4kevin/db/tech_st/11.1.0/dbs/APEX_XXX.DBF

Figure 8. Oracle Application Express Workspace Provisioning

The above steps will create a database schema, APEX_EBS_EXTENSION, with the correct permissions for building local database objects. However, if you are running Oracle E-Business Suite Release 12.2 or above, then you must update this schema to enable Edition-Based Redefinition (EBR). This is required as the APPS schema is enabled in order to achieve E-Business Suite Online Patching. Enable Edition-Based Redefinition by performing the following steps:

1. Connect to your Oracle E-Business Suite Database server
2. Start SQL*Plus and connect as SYS specifying the SYSDBA role
3. Alter the APEX schema:

   ```
   alter user APEX_EBS_EXTENSION enable editions;
   ```

{This functionality is incorporated into the apexebs_apex_setup.sql script referenced later in this paper}

If you are running Oracle E-Business Suite Release 12.2 or above then you will need to register this schema within E-Business Suite. Information on how to register the schema is available in the Oracle E-Business R12.2 Suite Maintenance Guide, Chapter 6 “Basic DBA Tasks”, section “Registering an ORACLE Schema”

Configuring Oracle Application Express in a Different Database

If you have installed Oracle Application Express into a different Oracle Database to the database where E-Business Suite is installed, then Oracle recommends ensuring the database link is closed correctly, by updating the Oracle REST Data Services configuration file utilized by Application Express. To update the configuration file perform these steps:

1. Connect to the server where Oracle Rest Data Services is configured
2. Navigate to the directory that holds the appropriate defaults.xml file
   {Such as /ords/config/apex}
3. Edit the defaults.xml file and add the following lines:

   ```
   <entry key="procedure.preProcess">apex_util.close_open_db_links
   </entry>
   <entry key="procedure.postProcess">apex_util.close_open_db_links
   </entry>
   ```

4. Restart Oracle Rest Data Services

You will also need to manually create the minimally privileged user, APEX_EBS_EXTENSION, within the E-Business Suite Oracle database. This schema will be used to access APPS objects, as it is very strongly discouraged to define a database link from another database directly to the APPS schema. It is also very strongly discouraged to define a database link from APPS to *any* other database. To create this user perform these steps:

1. Connect to your Oracle E-Business Suite Database server
2. Start SQL*Plus and connect as SYS specifying the SYSDBA role
3. Create the APEX_EBS_EXTENSION schema:

   ```
   create user APEX_EBS_EXTENSION
   identified by <<apex_ebs_extension>>
   default tablespace SYSAUX;
   ```

4. Grant the required rights to create database objects:

   ```
   grant create view to APEX_EBS_EXTENSION;
   ```

{This functionality is incorporated into the apexebs_apex_setup.sql script referenced later in this paper}

If you are running Oracle E-Business Suite Release 12.2 or above, then you must update this schema to enable Edition-Based Redefinition (EBR). This is required as the APPS schema is enabled in order to achieve E-Business Suite Online Patching. Enable Edition-Based Redefinition by performing the following steps:

1. Connect to your Oracle E-Business Suite Database server
2. Start SQL*Plus and connect as SYS specifying the SYSDBA role
3. Alter the APEX schema:

   ```
   alter user APEX_EBS_EXTENSION enable editions;
   ```

{This functionality is incorporated into the apexebs_apex_setup.sql script referenced later in this paper}

You should also create a workspace called APEX_EBS_LOCAL and a schema called APEX_EBS_LOCAL on the local database server where Oracle Application Express is configured. To create the workspace within the local installation of Application Express follow these steps:

1. Log into Oracle Application Express Administration (**http://<EBS_Hostname>:8080/apex/apex_admin**)
2. Enter *Username* **ADMIN**, *Password* (defined during installation by executing @apxchpwd), click Login
3. Navigate to Manage Workspaces > Create Workspace
4. Enter *Workspace Name* **APEX_EBS_LOCAL**, click *Next* >
5. Enter *Schema Name* **APEX_EBS_LOCAL**, and *Password* <<***apex_ebs_local>>***, click *Next* >
6. Enter *Administrator Username* **ADMIN**, *Administrator Password* <<***APEX_Password>>***, and your email address for *Email*, click *Next* >
7. Click Create Workspace, click *Done*

Now that you have the two schemas, APEX_EBS_EXTENSION within the E-Business Suite Oracle Database server and APEX_EBS_LOCAL in the Oracle Database where Application Express is configured, then you need to define the database link between these schema by following these steps:

1. Connect to your local Oracle Database server (where Application Express is configured)
2. Start SQL*Plus and connect as SYS specifying the SYSDBA role
3. Change the password for the APEX_EBS_LOCAL schema:

   ```
   alter user APEX_EBS_LOCAL identified by <<apex_ebs_local>>;
   ```

4. Connect to the EBS_APEX_LOCAL schema:

   ```
   connect APEX_EBS_LOCAL / <<apex_ebs_local>>
   ```

5. Add the database link:

   ```
   create database link APEX_EBS_DBLINK
   connect to APEX_EBS_EXTENSION
   identified by <<apex_ebs_extension>>
   using '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)
   (HOST=<<Remote_Host>>)(PORT=1521))(CONNECT_DATA=
   (SERVER=DEDICATED)(SERVICE_NAME=<<Remote_Service_Name>>)))'
   ```

   {*Remote_Host* and *Remote_Service_Name* refer to the Oracle Database where the APEX_EBS_EXTENSION schema is defined}

{This functionality is incorporated into the apexebs_apex_dblink_setup.sql script referenced later in this paper}

Configuring Developers within Application Express

Now that the workspace has been created, development will be performed within the Application Builder. To log into the Application Builder and create developers, follow these steps:

1. Log into Oracle Application Express Application Builder (**http://<EBS_Hostname>:8080/apex/apex**)
2. Enter *Workspace* **APEX_EBS_EXTENSION / APEX_EBS_LOCAL**, *Administrator Username* **ADMIN** and *Administrator Password* <<***APEX_Password>>*** entered when creating the workspace, click *Login*
3. You will need to reset the password – enter *Current Password* <<***APEX_Password>>***, *New Password* <<***Your New Password>>***, *Confirm New Password* <<***Your New Password>>***, click *Apply Changes*, click *Close Window*;
4. Navigate to Administration > Manage Users and Groups
5. Depending on your requirements you can create one or more developers using the Tasks listed on the right.

Accessing Oracle E-Business Suite Data

Oracle Application Express provides numerous wizards to rapidly build application components on existing Oracle tables or views. However, a prerequisite of many wizards within Oracle Application Express is that the underlying tables or views contain primary key constraints. Given Oracle E-Business Suite tables and views do not typically include primary key constraints, it is advisable to define additional database views that incorporate primary key constraints for the Oracle Applications objects you wish to access. For security purposes, it is recommended Oracle E-Business Suite data is accessed only through these views. The views can incorporate security, joins, etc., and prevent Oracle Application Express applications from making any unauthorized changes to the underlying data.

Below is sample code for creating such a view in the APPS schema –

```
CREATE OR REPLACE VIEW apex_ebs_user
    (  user_id
     , user_name
     , start_date
     , end_date
     , description
     , email_address
    , user_guid
    , person_party_id
    , CONSTRAINT APEX_EBS_USER_pk
           PRIMARY KEY (user_id)
           RELY DISABLE NOVALIDATE
    )
AS
    SELECT user_id
    ,      user_name
    ,      start_date
    ,      end_date
    ,      description
    ,      email_address
    ,      user_guid          /* Used for Single-Sign On */
    ,      person_party_id    /* FK to party information */
    FROM fnd_user;
```

It is then necessary to grant rights to the new schema (APEX_EBS_EXTENSION) from within the APPS schema -

```
GRANT SELECT ON apex_ebs_user TO APEX_EBS_EXTENSION;
GRANT SELECT on fnd_responsibility_vl TO APEX_EBS_EXTENSION;
```

The final step is to create a corresponding view within the APEX_EBS_EXTENSION schema -

```
CREATE OR REPLACE VIEW apex_ebs_user AS
  SELECT * from apps.apex_ebs_user;
```

If you have installed Application Express into a different Oracle Database to the database where E-Business Suite is installed, then you will need to create a corresponding view within the APEX_EBS_LOCAL schema that accesses the view in APEX_EBS_EXTENSION -

```
CREATE OR REPLACE VIEW apex_dblink_ebs_user AS
    SELECT * from apex_ebs_user@APEX_EBS_DBLINK;
```

Updating Oracle E-Business Suite Tables

Although it may be tempting to insert/update/delete records on Oracle E-Business Suite tables directly, this is unsupported and must be avoided. Please keep in mind that direct updates to Oracle E-Business Suite tables is not supported unless explicitly documented. Direct updates will bypass validation, security and business logic which could lead to data corruption or unexpected system behavior. Oracle recommends that all Oracle E-Business Suite updates are made through calls to public APIs. The Oracle E-Business Suite Integration Repository provides a searchable list of public APIs for the system.

Below is sample code for creating a package body within the APPS Schema:

```
CREATE OR REPLACE PACKAGE BODY apex_integration_sample_apis AS

FUNCTION apex_validate_login (  p_username   in  varchar2
                              , p_password   in  varchar2
                             ) RETURN BOOLEAN
IS
BEGIN
    RETURN fnd_user_pkg.validatelogin(p_username, p_password);
END apex_validate_login;

PROCEDURE apex_update_email (  p_username        in varchar2
                             , p_owner           in varchar2
                             , p_email_address   in varchar2
                            )
IS
BEGIN
    wf_event.setdispatchmode('async');
    fnd_user_pkg.updateuser
    (  x_user_name     => p_username
     , x_owner         => p_owner
     , x_email_address => p_email_address);
END apex_update_email;

END apex_integration_sample_apis;
/
```

When utilizing the Oracle Application Express wizards on the new views to define applications, DML automatic row processes will be defined referencing the view specified. These processes can be deleted and alternate processes written which call the appropriate APIs. An alternative is to create ‘INSTEAD OF’ triggers on these new views which call the API to change the data in the tables. These triggers only need to be defined once against each view and will allow the standard Oracle Application Express page processes that are generated as part of the page creation wizards to be utilized.

Below is sample code for creating such a trigger in the new schema. This code is not included in the sample code provided, as our examples don’t update the fnd_flex_values records.

```
CREATE OR REPLACE TRIGGER apex_fnd_flex_values_tr
    INSTEAD OF INSERT OR UPDATE ON apex_fnd_flex_values
DECLARE
    v_storage_value VARCHAR2(32000);
BEGIN
    IF INSERTING THEN
        fnd_flex_val_api.create_independent_vset_value
        (  p_flex_value_set_name => :NEW.flex_value_set_name
         , p_flex_value          => :NEW.flex_value
         , p_description         => :NEW.description
         , x_storage_value       => v_storage_value
        );
    ELSIF UPDATING THEN
        fnd_flex_val_api.update_independent_vset_value
        (  p_flex_value_set_name => :NEW.flex_value_set_name
         , p_flex_value          => :NEW.flex_value
         , p_description         => :NEW.description
         , x_storage_value       => v_storage_value
        );
    ENDIF;
END;
/
```

Downloading and Installing the Sample Code

The code samples given above are available for download from the Oracle Technology Network (OTN). To run these scripts, perform the following steps:

1. Download the APPS schema script file from OTN – http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/329274.sql
2. Save the file as **apexebs_apps_setup.sql** and upload to your Oracle E-Business Suite Database server
3. Connect to your Oracle E-Business Suite Database server
4. Start SQL*Plus and connect as SYS specifying the SYSDBA role
5. Run the script, providing the appropriate criteria: **`@apexebs_apps_setup`**
6. Download the APEX schema script file from OTN – http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/332106.sql
7. Save the file as **apexebs_apex_setup.sql** and upload to your Oracle E-Business Suite Database server
8. Connect to your Oracle E-Business Suite Database server
9. Start SQL*Plus and connect as SYS specifying the SYSDBA role
10. Run the script, providing the appropriate criteria: **`@apexebs_apex_setup`**

If you have installed Application Express into a different Oracle Database to the database where E-Business Suite is installed, then perform the following steps:

1. Download the APEX DBLINK schema script file from OTN – http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/2282591.sql
2. Save the file as **apexebs_apex_dblink_setup.sql** and upload to your Oracle E-Business Suite Database server
3. Connect to your local Oracle Database server (where Application Express is configured)
4. Start SQL*Plus and connect as the APEX Workspace schema [APEX_EBS_LOCAL]
5. Run the script, providing the appropriate criteria: **`@apexebs_apex_dblink_setup`**

Integrating with Oracle E-Business Suite

In order to connect from Oracle E-Business Suite to your new Oracle Application Express Applications, a secure mechanism is required to hand control from Oracle E-Business Suite to Oracle Application Express. The following actions need to be performed:

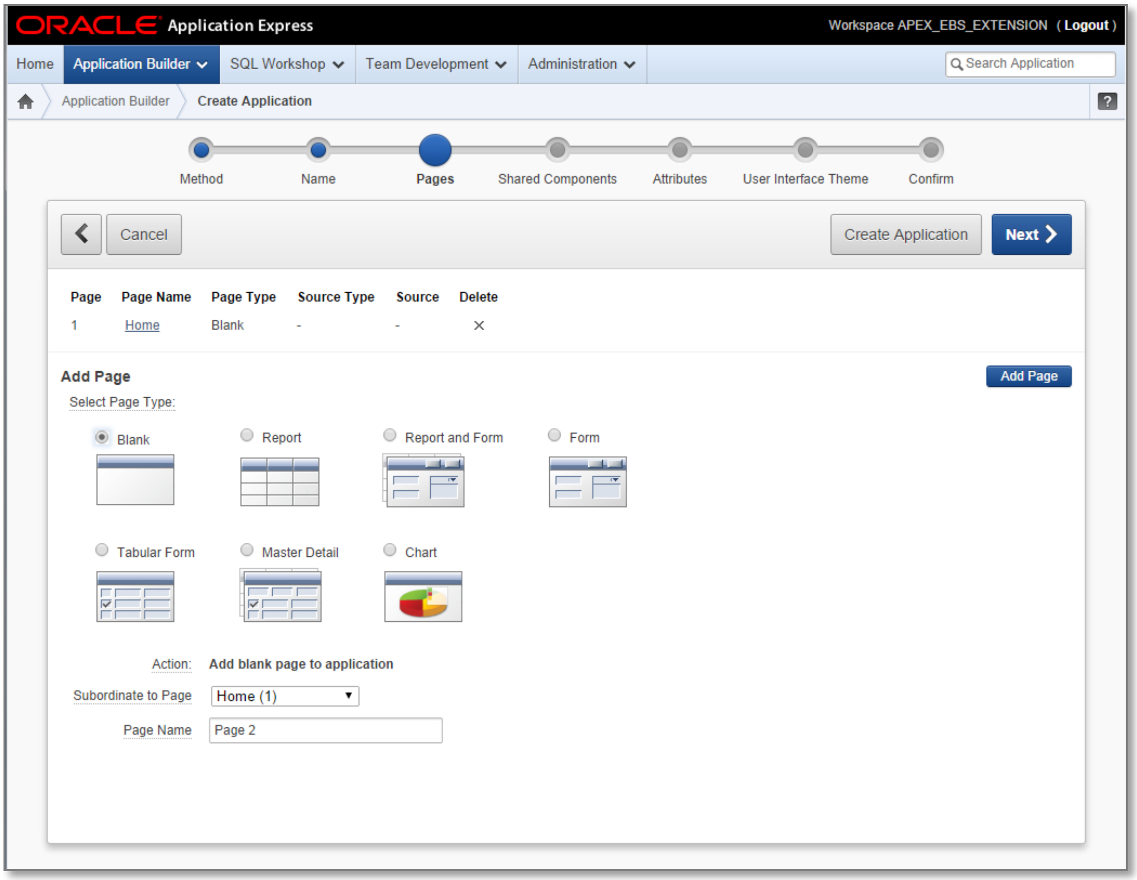
1. Create an Oracle Application Express application
2. Configure the Oracle Application Express login process
3. Define Oracle Application Express authorizations
4. Define Oracle E-Business Suite profile and functions
5. Link functions to menus and responsibility

Creating an Oracle Application Express Application

Developers define applications within Oracle Application Express as a collection of pages. Pages generally have one or more regions, buttons, items, computations, processes, branches, dynamic actions, tabs, lists, and breadcrumbs. Each application within an Oracle Application Express instance has a unique application ID. For end-users to access an Oracle Application Express application directly they enter a URL of the form: **http://<APEX_Host>:8080/apex/f?p=<Application_Id>:<Home_Page>**

To create an Oracle Application Express application, perform the following steps:

1. Navigate to the *Application Builder*
2. Click *Create*, select *Application Type* **Database**, click *From Scratch*
3. Enter *Name* **APEX EBS DEMO**, click *Next >*
4. Click *Create Application*



Page	Page Name	Page Type	Source Type	Source	Delete
1	Home	Blank	-	-	×

Select Page Type: ● Blank ○ Report ○ Report and Form ○ Form ○ Tabular Form ○ Master Detail ○ Chart

Action: Add blank page to application

Subordinate to Page: Home (1)

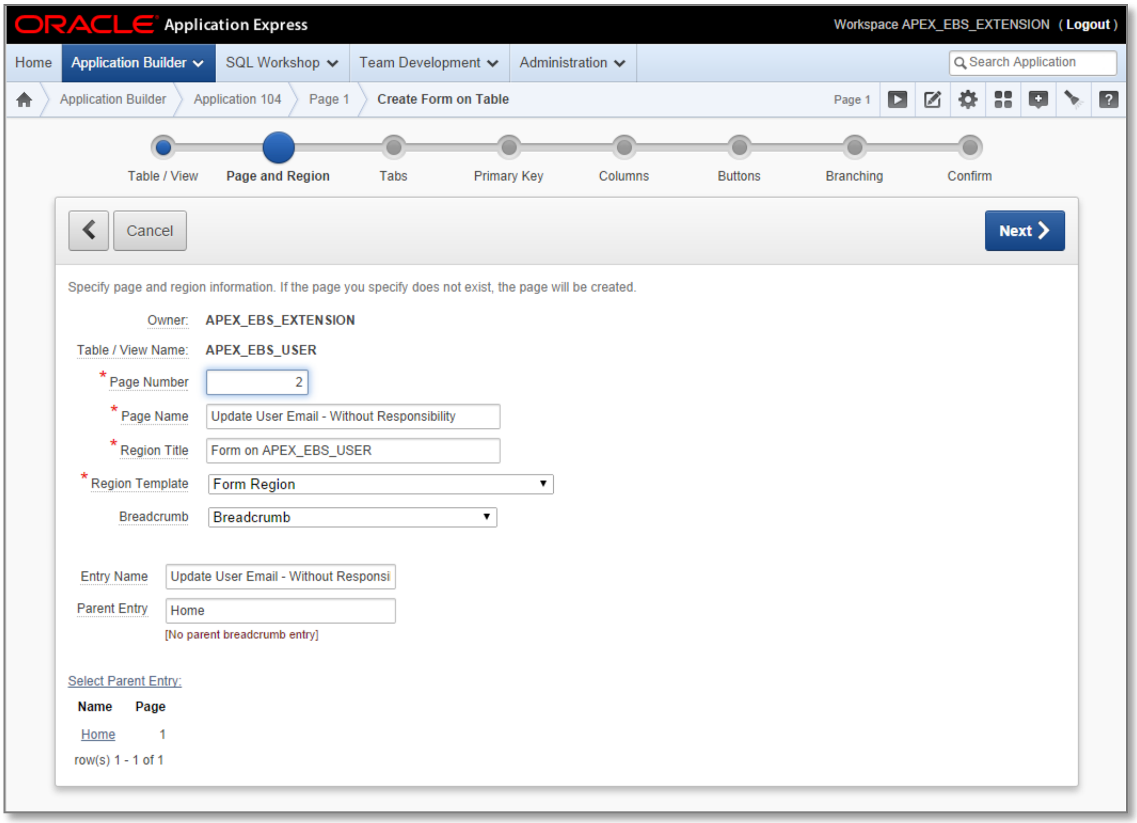
Page Name: Page 2

Figure 9. Creating an Oracle Application Express Application

Now we can add additional pages based on the APEX_EBS_USER view created earlier. We want to add two pages - one to be used for testing without responsibilities, and one for testing with Oracle E-Business Suite responsibilities. To add these pages perform the following steps:

1. Navigate to the Oracle Application Express Application Builder
2. Select the APEX EBS DEMO application
3. Click Create Page, select Form, select Form on a Table or View, click Next >
4. Enter/search for *Table/View Name* **APEX_EBS_USER**, click *Next >*
5. Enter *Page Number* **2**, *Page Name* and *Region Title* **Update User Email – Without Responsibility**, *Breadcrumb* **Breadcrumb**, *Parent Entry* **Home**, click *Next >*
6. Select *Tab Option* Use an existing tab set and create a new tab within the existing tab set, enter *New Tab Label* **Without Responsibility**, click *Next >*

7. Select *Primary Key Type* **Select Primary Key Column(s)**, select *Primary Key Column 1* **USER_NAME**, click *Next >*, click *Next >*
8. Choose all *Select Column(s)*, click *Next >*, click *Next >*
9. Enter *After Page Submit* and *When Cancel Button* **1**, click Next >
10. Click *Create*



Specify page and region information. If the page you specify does not exist, the page will be created.

Owner: APEX_EBS_EXTENSION

Table / View Name: APEX_EBS_USER

* Page Number: 2

* Page Name: Update User Email - Without Responsibility

* Region Title: Form on APEX_EBS_USER

* Region Template: Form Region

Breadcrumb: Breadcrumb

Entry Name: Update User Email - Without Responsi

Parent Entry: Home

[No parent breadcrumb entry]

Select Parent Entry:

Name	Page
Home	1

row(s) 1 - 1 of 1

Figure 10. Creating an Oracle Application Express Page

Update the generated *Fetch Row from APEX_EBS_USER* process (the process that retrieves the user details) to use the Oracle Application Express user name:

1. From the Application Builder for the APEX EBS Demo application - Edit Page 2
2. Double-click the *Fetch Row from APEX_EBS_USER* process
3. Enter Item Containing Primary Key Column Value **APP_USER**, click **Apply Changes**

The items on the page were generated as text items by default. For this example we only want the user to be able to update the Email Address so we need to update the page and change item properties:

1. From the Application Builder for the APEX EBS Demo application - Edit Page 2
2. Double-click the *User Name* item
3. Select *Display As* **Display Only**, click > {Displays P2_START_DATE}
4. Select *Display As* **Display Only**, click > {Displays P2_END_DATE}
5. Select *Display As* **Display Only**, click > {Displays P2_DESCRIPTION}
6. Select *Display As* **Display Only**, click > {Displays P2_EMAIL_ADDRESS}
7. Click > {Displays P2_PERSON_PARTY_ID}
8. Select *Display As* **Display Only**, click *Apply Changes*

Delete the generated *Process Row of APEX_EBS_USER* process (the process that updates the user details) and replace it with a call to the *apex_update_email* procedure defined earlier:

1. From the Application Builder for the APEX EBS Demo application - Edit Page 2
2. Double-click the *Process Row of APEX_EBS_USER* process
3. Click *Delete*, click *Ok*
4. Right-click on *Processes* region, click *Create*
5. Select *Category* **PL/SQL**, click *Next* >
6. Enter *Name* **Update Email**, click *Next* >
7. Enter *PL/SQL Page Process*:

   ```
   apps.apex_integration_sample_apis.apex_update_email
     (  p_username      => :APP_USER
      , p_owner         => :APP_USER
      , p_email_address => :P2_EMAIL_ADDRESS
     );
   ```

8. Click *Next* >
9. Enter *Success Message* **Email updated successfully**, *Failure Message* **Email not updated**, click *Next* >
10. Select *When Button Pressed* **SAVE (Apply Changes)**, click *Create Process*

Repeat the same steps as above to create another page, Page 3, with the same details except *Page Title* **Update User Email – Using Responsibility**, *New Tab Label* **Using Responsibility**. Update the *Fetch Row from APEX_EBS_USER* process, replace the *Process Row of APEX_EBS_USER with* Update Email, and set all Page 3 items to be **Display Only** except for P3_USER_ID (Hidden) and P3_EMAIL_ADDRESS (Text Field).

To show the Oracle E-Business Suite responsibility that is going to be passed to Page 3 you must add a new process, region, and two items, using the following steps:

1. From the Application Builder for the APEX EBS Demo application - Edit Page 3
2. Create a new region – Right click on *Body (3)*, click *Create*
3. Select *Type of Region* **HTML**, click *Next* >
4. Enter *Title* **Responsibility**, click *Next* >, click *Create Region*
5. Create Items – Right click on *Responsibility* region, click *Create Page Item*
6. Select *Item Type* **Display Only**, enter *Item Name* **P3_RESPONSIBILITY_NAME**, click *Next* >, click *Next* >, click *Next* >,click *Create Item*
7. Right click on *Responsibility* region, click *Create Page Item*
8. Select *Item Type* **Display Only**, enter *Item Name* **P3_RESP_DESCRIPTION**, click *Next* >
9. Enter *Label* **Description**, click *Next* >, click *Next* >,click *Create Item*
10. Click on Create > Page control on this page, select Control Type **Process**, select Category **PL/SQL**, click Next >
11. Enter *Name* **Fetch Responsibility**, *Point* On Load – After Header, click *Next* >
12. Enter *PL/SQL Page Process*:

```
begin
  for c1 in (select responsibility_name
             ,      description
             from apps.fnd_responsibility_vl
             where application_id    = :EBS_APP_ID
             and   responsibility_id = :EBS_RESP_ID
            ) loop
    :P3_RESPONSIBILITY_NAME := c1.responsibility_name;
    :P3_RESP_DESCRIPTION    := c1.description;
  end loop;
exception
  when others then
    null;
end;
```

13. Click *Create Process*

To further aid navigation on the Home page you can add icons to go to each of the pages:

From the Application Builder for the APEX EBS Demo application – go to Shared Components

1. Click *Lists*, click *Create*, *Name* **Responsibility**, click *Next* >
2. Enter two list entries - *List Entry Label* **Without Responsibility**, *Target Page ID* **2**; *List Entry Label* **Using Responsibilities**, *Target Page ID* **3**, click *Next* >, click *Create List*
3. Click on Shared Components > Lists, double-click Responsibility
4. Click *Without Responsibility*, enter *Image* **menu/address_book_bx_128x128.png**, click > {Displays *Using Responsibility*}
5. Enter *Image* **menu/addresses_bx_128x128.png**, click *Apply Changes*
6. Navigate to Page 1 {in Application Builder}
7. Create a new region – Right click on *Regions*, click *Create*
8. Select *Type of Region* **List**, click *Next* >
9. Enter *Title* **Responsibility**, *Region Template* **DIV Region with ID**, click *Next* >
10. Select *List* **Responsibility**, *List Template* **Horizontal Images with Label List**, click *Create List Region*.

At this stage the application is using default Oracle Application Express user credentials. Run the application and enter the user name and password you defined when creating the APEX_EBS_EXTENSION Workspace earlier:

1. From the Application Builder for the APEX EBS Demo application – click *Run*
2. Enter *Username* **ADMIN**, *Password* **APEX_Password1**, click *Login*
3. Navigate using the icons and tabs

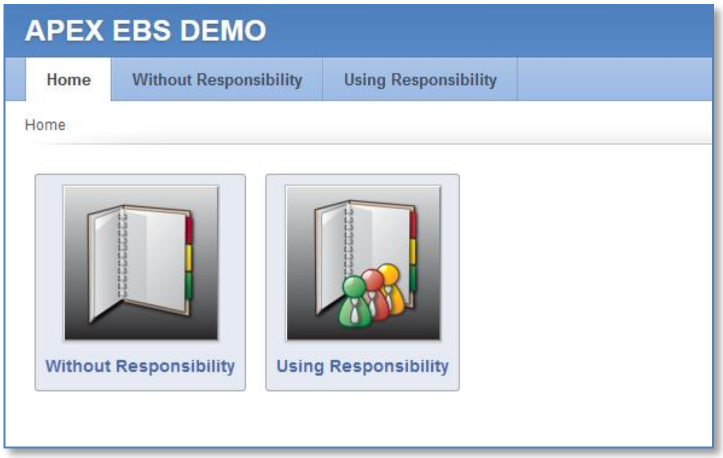


Figure 11. Oracle Application Express application

Note: No data will be displayed on Pages 2 or 3 because user credentials will not be found in the APEX_EBS_USER view (because you are currently using Oracle Application Express credentials instead of Oracle E-Business Suite credentials). Later in the document we will define authentication schemes that allow log in using the E-Business Suite credentials and that user's data will be displayed.

Oracle Application Express Authentication and Authorization

Oracle Application Express provides “out-of-the-box” mechanisms to handle both authentication (Is the user a valid Oracle E-Business Suite user?) and authorization (What privileges / responsibility does the user have within the application?).

Authentication schemes check the user’s user name and password credentials before the user is allowed to access the application. Authorization schemes, on the other hand, control display and user access to pages, regions, items, buttons, and processes within an application.

If you are using Oracle Application Express Release 4.1 or above then you can also define plug-in authentications and plug-in authorizations. The benefit of developing such plug-ins is to ensure consistency across different applications. [See: Advanced Programming Techniques – Implementing Plug-Ins http://docs.oracle.com/cd/E37097_01/doc/doc.42/e35125/advnc_plugins.htm#BACGADAG].

User Authentication

There are several pre-configured authentication schemes defined within Oracle Application Express that can be utilized, specifically Oracle Single Sign-On (OSSO), HTTP Header Variable (available with Oracle Application Express Release 4.1 and above) for use with Oracle Access Manager (OAM), or custom authentication schemes [See: Managing Application Security – Establishing User Identity Through Authentication http://docs.oracle.com/cd/E37097_01/doc/doc.42/e35125/sec_authentication.htm#BABHIEIA].

Once you have configured OAM, OSSO or a custom authentication, you will be able to log into your Oracle Application Express applications using any valid OAM, OSSO or Oracle E-Business Suite user name and password.

Configuring the Oracle Application Express Login Process

If using custom authentication, users will not automatically log into Oracle Application Express. However, if you are using Oracle Access Manager or Oracle Single Sign-On then users will not be required to enter their user credentials when Oracle Application Express is invoked from the Oracle E-Business Suite menu.

Configuring Oracle Access Manager

Many Oracle E-Business Suite installations use Oracle Access Manager (OAM) or Oracle Single Sign-On (OSSO) to provide a centralized method of authenticating users. The Oracle standard for single user authentication with Oracle E-Business Suite Release 12 is Oracle Access Manager.

If you are using OAM then follow the Integrating Oracle Application Express with Oracle Access Manager white paper available from the Oracle Technology Network (OTN): http://www.oracle.com/technetwork/developer-tools/apex/learnmore/apex-oam-integration-1375333.pdf. You will need to integrate Oracle Access Manager with Application Express separately from integration with E-Business Suite, but they can both use the same components. Once they are both configured then they can utilize the same user authentication.

Configuring Single Sign-On

If you are using OSSO with your Oracle E-Business Suite, you will have to identify the Oracle Application Express engine as a partner application within the OSSO infrastructure. Once you have configured the OSSO infrastructure, you can simply create an Oracle Application Express authentication scheme using the name given to the OSSO entry to integrate user authentication with your Oracle E-Business Suite environment. An OSSO-based authentication scheme will use the OSSO login page for users of your Oracle Application Express application. This method has the downside that a new session will be established each time Oracle Application Express is called from an Oracle E-Business Suite function, thereby losing any session information previously established. Application developers should keep this in mind when developing applications.

Before configuring Single Sign-On you will need the following:

- Download the appropriate OSSO SDK (Software Development Kit)
- Details regarding the OSSO Server – Site Id, Site Token, Encryption Key, Single Sign-On URL, Single Sign-Off URL
- A copy of **custom_auth_sso_902.sql** and **custom_auth_sso_902.plb** from the Oracle Application Express installation – ***apex/core*** directory

To configure Oracle Application Express as a partner application within OSSO perform the following steps:

1. Copy the SSO SDK to the Oracle E-Business Suite Database Server
2. Unzip the SSO SDK file
3. Copy **custom_auth_sso_902.sql** and **custom_auth_sso_902.plb** to the ***packages*** directory (created when the SSO SDK file was unzipped)
4. Connect to your Oracle E-Business Suite Database server and navigate to the ***packages*** directory
5. Run SQL*Plus, connect as SYS AS SYSDBA – **sqlplus / AS SYSDBA**
   (All following statements will be run from the SQL prompt; Replace all passwords with a case sensitive password which is not the same as the user being altered)

```
alter user APEX_040200 unlock identified by <APEX_040200_Password>;

alter user APEX_PUBLIC_USER unlock identified by
<APEX_PUBLIC_USER_Password>;

alter user APEX_EBS_EXTENSION identified by
<APEX_EBS_EXTENSION_Password>;

connect APEX_EBS_EXTENSION/<APEX_EBS_EXTENSION_Password>

@loadsdk.sql

@regapp.sql  (The Listener Token will be HTML_DB:<EBS_Hostname>:8080 ,
specify the Partner App Name APEX_EBS_EXTENSION_SSO; Use the values
determined above for Site Id, Site Token, etc.)

grant execute on wwsec_sso_enabler_private to APEX_040200;

connect APEX_040200/<APEX_040200_Password>

@custom_auth_sso_902.sql

@custom_auth_sso_902.plb

grant execute on wwv_flow_custom_auth_sso to APEX_PUBLIC_USER;

connect APEX_PUBLIC_USER/<APEX_PUBLIC_USER_Password>

create public synonym wwv_flow_custom_auth_sso for
apex_040200.wwv_flow_custom_auth_sso

exit
```

Once you have configured the Single Sign-On you will now need to define the authentication scheme within Oracle Application Express. Follow these steps to create your SSO authentication:

1. Log into Oracle Application Express Application Builder (**http://<EBS_Hostname>:8080/apex/apex**)
2. From the Application Builder for the APEX EBS Demo application – go to Shared Components
3. Click *Authentication Schemes*, click *Create*, select **Based on a pre-configured scheme from the gallery**, click *Next >*
4. Select Oracle Application Server Single Sign-On (My Application as Partner App), click
   Next >
5. Enter *Partner Application Name* **APEX_EBS_EXTENSION_SSO**, click *Next >*
6. Enter *Name* **EBS SSO Login**, click *Create Scheme*
7. Click Change Current Tab, select Available Authentication Schemes **EBS SSO LOGIN**, click Next >, click Make Current
8. Click Run Application

If correctly configured, when you run the application, you will be redirected to the OSSO Login Page and can enter your Single Sign-on credentials to log into Oracle Application Express.

Configuring Custom Authentication

If your Oracle E-Business Suite installation does not use Oracle Access Manager or Oracle Single Sign-On, you can create a custom authentication scheme which calls a PL/SQL function within Oracle Application Express to validate the identity of the user, based on the password sent.

Below is sample code for creating a function to determine if the user if valid –

```
CREATE OR REPLACE FUNCTION apex_validate_login
    (  p_username IN VARCHAR2
     , p_password IN VARCHAR2
    ) RETURN BOOLEAN
AS
BEGIN
    RETURN fnd_user_pkg.validatelogin(p_username, p_password);
END validate_login;
```

Note: This specific code is included in the **apexebs_apps_setup.sql** file you downloaded and installed into the APPS schema earlier in this paper. This code is included within that file in *Package* **apex_integration_sample_apis**, *Function* **apex_validate_login**. This package function is used below when defining a custom authentication.

Once you have created this function in your APPS schema you need to define the authentication scheme within Oracle Application Express. Follow these steps to create your custom authentication:

1. Log into Oracle Application Express Application Builder (**http://<EBS_Hostname>:8080/apex/apex**)
2. From the Application Builder for the APEX EBS Demo application – go to Shared Components
3. Click *Authentication Schemes*, click *Create*, select **Based on a pre-configured scheme from the gallery**, click *Next* >
4. Enter *Name* **EBS Custom Login**, select *Schema Type* **Custom,** enter *Authentication Function Name* **apps.apex_integration_sample_apis.apex_validate_login**, click *Create Authentication Scheme*
5. Click Run Application
6. Enter Oracle E-Business Suite credentials – *Username* **APEX**, *Password* <<***Welcome1>>***

Defining Oracle Application Express Authorizations

Authentication schemes only check that the user has a valid Oracle E-Business Suite login, not that the user is authorized to use an application. Oracle Application Express provides the ability to define authorization schemes that can be used on every component within an application (for example pages, regions, buttons, items, validations, etc.) to restrict access [See: Managing Application Security – Providing Security Through Authorization http://docs.oracle.com/cd/E37097_01/doc/doc.42/e35125/sec_authorization.htm#BABEDFGB].

Oracle E-Business Suite includes the concept of responsibilities, which determine what capabilities and data users are allowed to access. The recommended practice is to create Oracle Application Express authorization schemes that mimic the necessary Oracle E-Business Suite responsibilities to define cohesive access plans.

It is very strongly recommended that authorizations are applied to all non-public Oracle Application Express pages (and those limited by Oracle E-Business Suite responsibilities) rather than just to menu items. Session state protection should also be defined for the application to prevent users from accessing pages by manipulating the URL [See: Managing Application Security – Understanding Session State Protection http://docs.oracle.com/cd/E37097_01/doc/doc.42/e35125/sec_dev.htm#CDDGIGJH].

Using Oracle E-Business Suite Responsibilities with Integrated Access

By modifying the function call used to invoke Oracle Application Express from Oracle E-Business Suite you can pass the parameters required to test the responsibility directly. By defining an Oracle Application Express authorization scheme which first performs an APPS_INITIALIZE and then returns the result from FND_FUNCTION.TEST you can use this authorization to prevent unauthorized access to pages and processes to users without the correct responsibility.

In order to call APPS procedures and functions from another schema it is necessary to define a package with DEFINER authorization such that the PL/SQL is run with the privileges of the APPS schema rather than the privileges of the Oracle Application Express schema.

Below is sample code for creating such a DEFINER Package in the APPS schema –

```
CREATE OR REPLACE PACKAGE apex_global AUTHID DEFINER AS
  PROCEDURE apps_initialize(
     user_id IN NUMBER,
     resp_id IN NUMBER,
     resp_appl_id IN NUMBER,
     security_group_id IN NUMBER DEFAULT 0,
     server_id IN NUMBER DEFAULT -1);
  FUNCTION function_test(function_name IN VARCHAR2) RETURN BOOLEAN;
end;
/

CREATE OR REPLACE PACKAGE BODY apex_global AS
  PROCEDURE apps_initialize(
    user_id IN NUMBER,
    resp_id IN NUMBER,
    resp_appl_id IN NUMBER,
    security_group_id IN NUMBER DEFAULT 0,
    server_id IN NUMBER DEFAULT -1) IS
  BEGIN
    fnd_global.apps_initialize(user_id, resp_id, resp_appl_id,
                               security_group_id, server_id);
  END;

FUNCTION function_test(function_name IN VARCHAR2) RETURN BOOLEAN IS
  BEGIN
    RETURN fnd_function.test(function_name);
  END;
END;
/
```

Note: This specific code is included in the **apexebs_apps_setup.sql** file you downloaded and installed into the APPS schema earlier in this paper.

You need to create three Oracle Application Express - Application Items to hold the Oracle E-Business Suite Responsibility Id, Application Id, and Security Group Id parameters that are being passed by the Oracle E-Business Suite function APEX_DEMO_2. Follow these steps to add Application Items into Oracle Application Express:

1. Log into Oracle Application Express Application Builder (**http://<EBS_Hostname>:8080/apex/apex**)
2. From the Application Builder for the APEX EBS Demo application – go to Shared Components
3. Click Application Items, click Create >, enter Name **EBS_RESP_ID**, click Create Application Item
4. Click *Create* >, enter *Name* **EBS_APP_ID**, click *Create Application Item*
5. Click *Create* >, enter *Name* **EBS_SEC_GROUP**, click *Create Application Item*

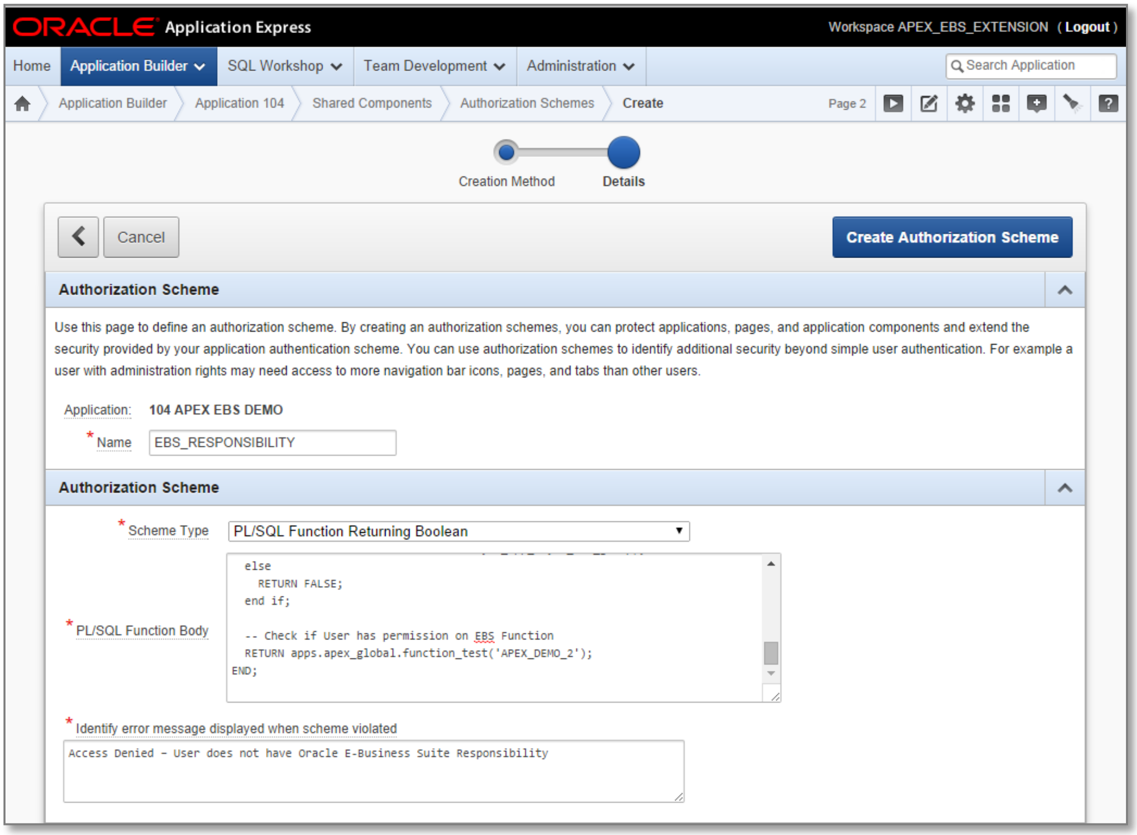
Below is sample code for defining an Oracle Application Express authorization –

```
DECLARE
  l_user_id   NUMBER DEFAULT 0;
  l_resp_id   NUMBER DEFAULT 0;
  l_app_id    NUMBER DEFAULT 0;
  l_sec_group NUMBER DEFAULT 0;
BEGIN
  for c1 in (select user_id from APEX_EBS_USER
              where user_name = :APP_USER
                ) loop
      L_user_id := c1.user_id;
  end loop;
  begin
    select nvl(to_number(:EBS_RESP_ID),0) into l_resp_id from dual;
    select nvl(to_number(:EBS_APP_ID),0) into l_app_id from dual;
    select nvl(to_number(:EBS_SEC_GROUP),0) into l_sec_group from dual;
  exception
    when others then
      l_resp_id := 0;
      l_app_id    := 0;
      l_sec_group := 0;
  end;
  -- Call Initialize with the User (based on APEX log in) and the
  -- Responsibility Id, Account Id, and Security Group if passed from
EBS
  if l_resp_id <> 0 then
     apps.apex_global.apps_initialize(  l_user_id, l_resp_id
                                      , l_app_id, l_sec_group);
  else
    RETURN FALSE;
  end if;

  -- Check if User has permission on EBS Function
  RETURN apps.apex_global.function_test('APEX_DEMO_2');
END;
```

Follow these instructions to create the Oracle Application Express authentication scheme and restrict Page 3:

1. From the Application Builder for the APEX EBS Demo application – go to Shared Components
2. Click Authorization Schemes, click Create >, select Create Authorization Scheme **From Scratch**, click Next >
3. Enter *Name* **EBS_RESPONSIBILITY**, select *Schema Type* **PL/SQL Function Returning Boolean**, copy the SQL code above into *Expression 1*, enter *Identify error message* **Access Denied – User does not have Oracle E-Business Suite Responsibility**, click *Create Authorization Scheme*



Authorization Scheme

Use this page to define an authorization scheme. By creating an authorization schemes, you can protect applications, pages, and application components and extend the security provided by your application authentication scheme. You can use authorization schemes to identify additional security beyond simple user authentication. For example a user with administration rights may need access to more navigation bar icons, pages, and tabs than other users.

Application:	104 APEX EBS DEMO
* Name	EBS_RESPONSIBILITY

Authorization Scheme

* Scheme Type	PL/SQL Function Returning Boolean

* PL/SQL Function Body

```
  else
    RETURN FALSE;
  end if;

  -- Check if User has permission on EBS Function
  RETURN apps.apex_global.function_test('APEX_DEMO_2');
END;
```

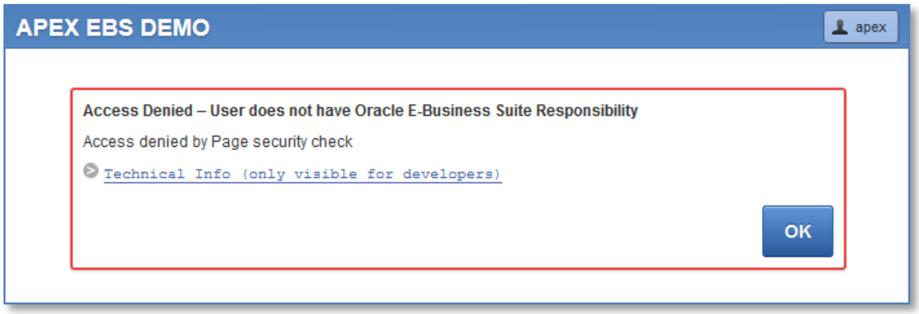
* Identify error message displayed when scheme violated

```
Access Denied - User does not have Oracle E-Business Suite Responsibility
```

Figure 12. Defining Oracle Application Express Authorization Scheme

4. From the Application Builder for the APEX EBS Demo application - Edit Page 3
5. Double-click Update User Email – Using Responsibility {Page Title}
6. Select *Security* > *Authorization Scheme* **EBS_RESPONSIBILITY**, click Apply Changes
7. From the Application Builder for the APEX EBS Demo application – go to Page 1

8. Run Application
9. Click Using Responsibilities – Access is denied due to the authorization.



Access Denied – User does not have Oracle E-Business Suite Responsibility

Access denied by Page security check

Figure 13. Authorization error message

Installing the Sample Packaged Application

To quickly configure Oracle Application Express within your Oracle E-Business Suite environment, a sample Oracle Application Express application is available which includes working examples for updating the user's email address based on the steps performed previously in this paper.

To download from Oracle Technology Network (OTN) and install the sample application perform the following steps:

1. Download the Oracle Application Express Sample Application script file from OTN – http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/332107.sql
2. Save the file as **apexebs_apex_application_rev2.sql** on your local desktop
3. Log into Oracle Application Express Application Builder (**http://<EBS_Hostname>:8080/apex**)
4. From the Application Builder – click *Import*
5. Select *Import File* **apexebs_apex_application_rev2.sql**, click *Next* >, click *Next* >, click *Install*
6. Click Run Application

Note: This application will produce errors at runtime unless the **apexebs_apps_setup.sql** script has been run in the APPS schema and the **apexebs_apex_setup.sql** script has been run in the APEX_EBS_EXTENSION schema as outlined earlier in this paper.

Sample Packaged Application in a Different Database

If you have installed Oracle Application Express into a different Oracle Database to the database where E-Business Suite is installed, then the application above needs to be modified to access the appropriate view, namely `APEX_DBLINK_EBS_USER`, and all calls to APPS need to be suffixed with the database link name, namely `@APEX_EBS_DBLINK`. You can also download the modified application from Oracle Technology Network (OTN) using these steps:

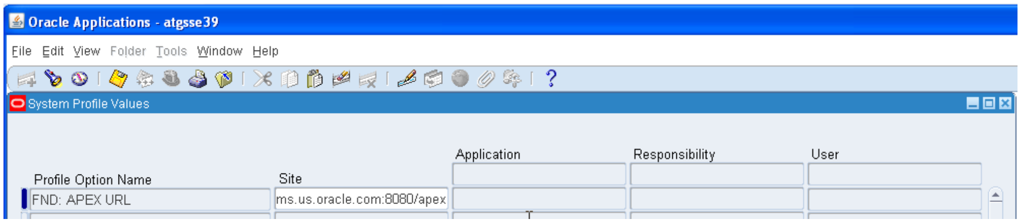
1. Download the Oracle Application Express Sample DB Link Application script file from OTN http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/2283639.sql
2. Save the file as **apexebs_apex_dblink_app_rev2.sql** on your local desktop
3. Log into Oracle Application Express Application Builder (**http://<Local APEX Home>:8080/apex**)
4. From the Application Builder – click *Import*
5. Select *Import File* **apexebs_apex_dblink_app_rev2.sql**, click *Next* >, click *Next* >, click *Install*
6. Click Run Application

Defining the Oracle E-Business Suite Profile and Functions

With Oracle E-Business Suite 11i you can redirect to Oracle Application Express by creating a web-enabled mod_plsql procedure. However, with Oracle E-Business Suite Release 12 this integration must be implemented on the Oracle E-Business Suite Application Server as a Java Server Page (JSP). Oracle E-Business Suite provides a way of configuring URLs to Oracle Application Express and will create the necessary JSP based on the defined parameters. This configuration requires setting an Oracle E-Business Suite profile option (FND: APEX URL) to contain the Oracle REST Data Services host name, and defining Oracle E-Business Suite functions that point to the actual APEX pages.

Update the FND: APEX URL profile option with the correct setting at the site level using the following steps:

1. Log in to Oracle E-Business Suite with the SYSADMIN user (**http://<EBS_Hostname>:8074/OA_HTML/AppsLogin**)
2. Navigate to the System Administrator responsibility > Profile > System menu option
3. Search for *Profile* **%APEX%**, click *Find*
4. For profile FND: APEX URL enter *Site* **http://<EBS_Hostname>:8080/apex**
5. Save the profile


Oracle Applications - atgsse39
File Edit View Folder Tools Window Help
System Profile Values

Profile Option Name	Site	Application	Responsibility	User
FND: APEX URL	ms.us.oracle.com:8080/apex			

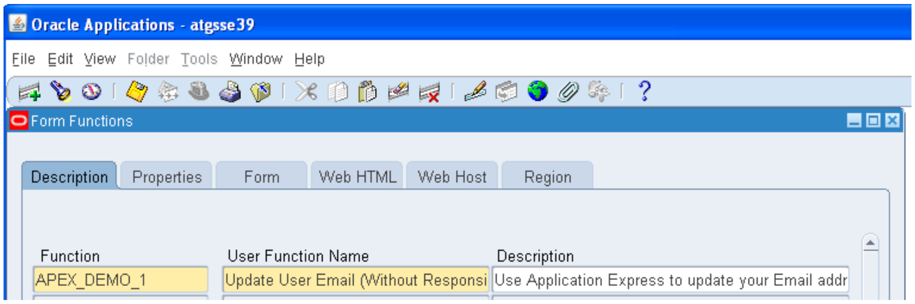
Figure 14. Updating Oracle E-Business Suite profile

Perform the following steps to define the Oracle E-Business Suite functions:

1. Navigate to the System Administrator responsibility > Application > Function menu option
2. For calls to the page without responsibility, create a function with the following details:

   Description -

   Function: **APEX_DEMO_1**
   User Function Name: **Update User Email (Without Responsibility)**

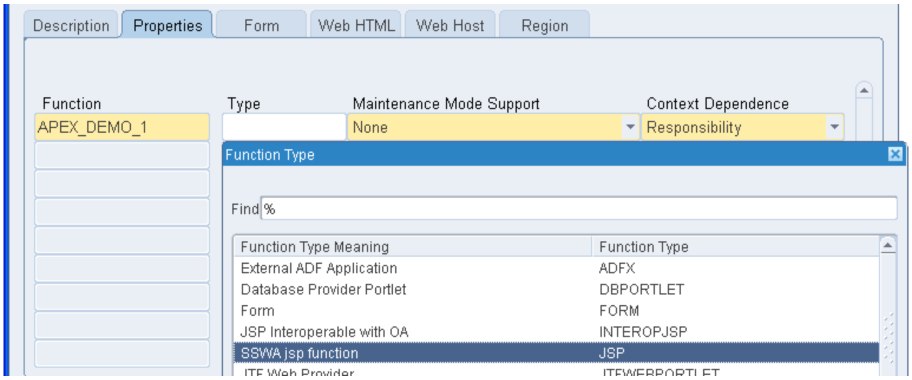

Oracle Applications - atgsse39
File Edit View Folder Tools Window Help
Form Functions
Description
Properties
Form
Web HTML
Web Host
Region

Function	User Function Name	Description
APEX_DEMO_1	Update User Email (Without Responsi	Use Application Express to update your Email addr

Figure 15. Creating Oracle E-Business Suite Function – Description Tab

Properties –

Type: **JSP**



Function	Type	Maintenance Mode Support	Context Dependence
APEX_DEMO_1		None	Responsibility

Find %

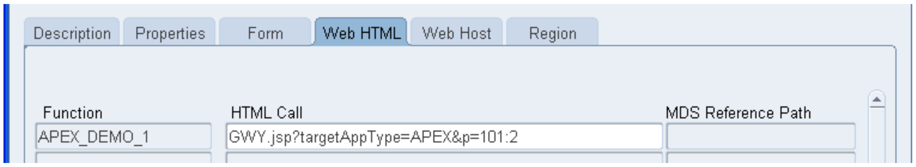
Function Type Meaning	Function Type
External ADF Application	ADFX
Database Provider Portlet	DBPORTLET
Form	FORM
JSP Interoperable with OA	INTEROPJSP
SSWA jsp function	JSP
ITF Web Provider	ITFWEBPORTLET

Figure 16. Creating Oracle E-Business Suite Function – Properties Tab

Web HTML –

HTML Call: **GWY.jsp?targetAppType=APEX& p=<APEX Application Id>:<APEX Page>:<Session>:<Request>:<Debug>:<Clear Cache>:<Parameter Pairs>**

{For example, to call Oracle Application Express application 109, Page 2 use `GWY.jsp?targetAppType=APEX&p=109:2`, all other parameters are optional}



Function	HTML Call	MDS Reference Path
APEX_DEMO_1	GWY.jsp?targetAppType=APEX&p=101:2	

Figure 17. Creating Oracle E-Business Suite Function – Web HTML Tab

3. For calls to the page with responsibility create a function with the following details -
   Function: **APEX_DEMO_2**
   User Function Name: **Update User Email (Using Responsibilities)**
   Type: **JSP**
   HTML Call: **GWY.jsp?targetAppType=APEX& p=<APEX Application Id>:<APEX Page>:<Session>:<Request>:<Debug>:<Clear Cache>:EBS_RESP_ID,EBS_APP_ID,EBS_SEC_GROUP:[RESPONSIBILITY_ID],[RESP_APPL_ID],[SECURITY_GROUP_ID]**
   {For example, to call Oracle Application Express application 109, Page 3 use **GWY.jsp?targetAppType=APEX&p=109:3:::::EBS_RESP_ID, EBS_APP_ID, EBS_SEC_GROUP:[RESPONSIBILITY_ID],[RESP_APPL_ID], [SECURITY_GROUP_ID]**}

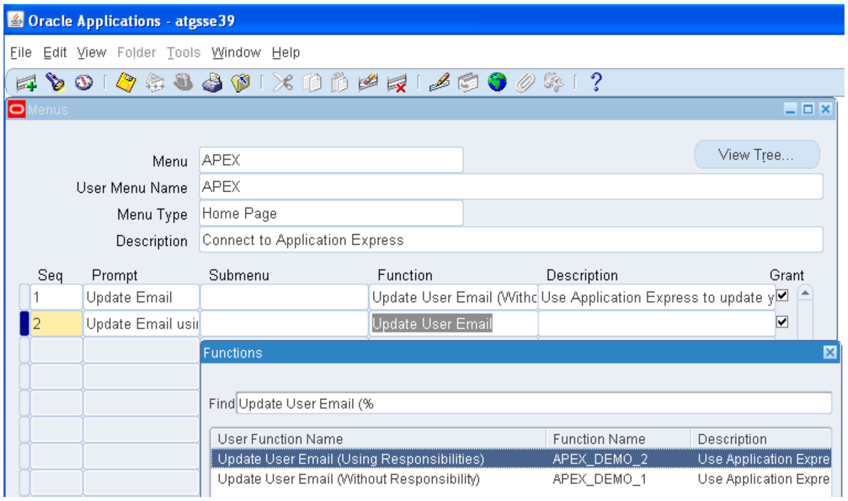
4. Save the functions

Defining Oracle E-Business Suite Menus and Responsibilities

The next step is to define menu options and responsibilities for your Oracle Application Express applications within the Oracle E-Business Suite environment. Once the function is defined you may attach it to appropriate Menus and Responsibilities. Users having access to such responsibilities will be able to see the link in the Oracle E-Business Suite home page. Once they click the link the current browser will launch the targeted APEX page in Oracle Application Express.

Perform the following steps to add an Oracle E-Business Suite menu:

1. Navigate to the System Administrator responsibility > Application > Menu option
2. Create a new menu -
   Name: **APEX**
   User Menu Name: **APEX**
   Menu Type: **Home Page**
3. Create menu component –
   Sequence: **1**
   Prompt: **Email Update (Without Responsibility)**
   Function: **APEX_DEMO_1** {Search for **Update User Email%**}
4. Create menu component –
   Sequence: **2**
   Prompt: **Email Update (Using Responsibilities)**
   Function: **APEX_DEMO_2** {Search for **Update User Email%**}
5. Save the menu


Oracle Applications - atgsse39
File Edit View Folder Tools Window Help
Menus
View Tree...
Functions

Menu	APEX
User Menu Name	APEX
Menu Type	Home Page
Description	Connect to Application Express

Seq	Prompt	Submenu	Function	Description	Grant
1	Update Email		Update User Email (Withc	Use Application Express to update y	☑
2	Update Email usi		Update User Email		☑

Find Update User Email (%

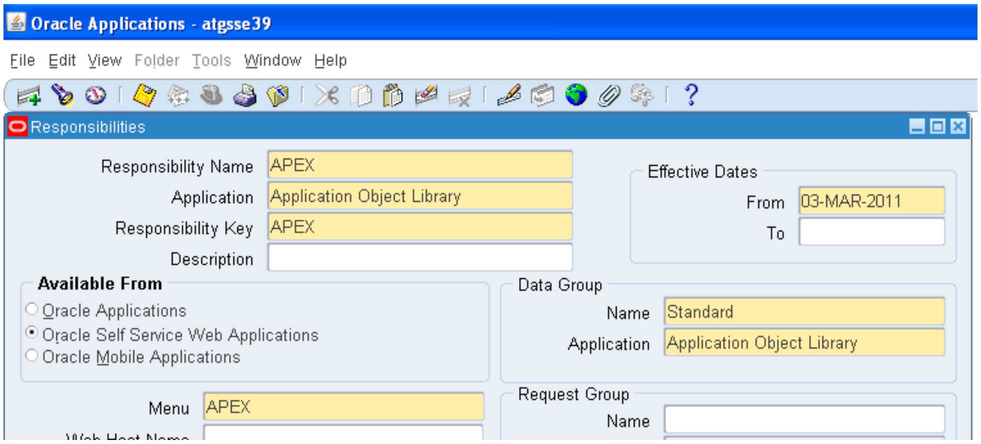
User Function Name	Function Name	Description
Update User Email (Using Responsibilities)	APEX_DEMO_2	Use Application Expre
Update User Email (Without Responsibility)	APEX_DEMO_1	Use Application Expre

Figure 18. Creating Oracle E-Business Suite Menu

Perform the following steps to add an Oracle E-Business Suite responsibility:

1. Navigate to the System Administrator responsibility > Security > Responsibilities >Define option
2. Create a new responsibility -

   Responsibility Name: **APEX**
   Application: **Application Object Library**
   Responsibility Key: **APEX**
   Available From: **Oracle Self Service Web Applications**
   Data Group: **Standard**
   Application: **Application Object Library**
   Menu: **APEX**
3. Save the responsibility



Responsibility Name: APEX
Application: Application Object Library
Responsibility Key: APEX
Description:

Effective Dates
From: 03-MAR-2011
To:

Available From
○ Oracle Applications
● Oracle Self Service Web Applications
○ Oracle Mobile Applications

Data Group
Name: Standard
Application: Application Object Library

Menu: APEX

Request Group
Name:

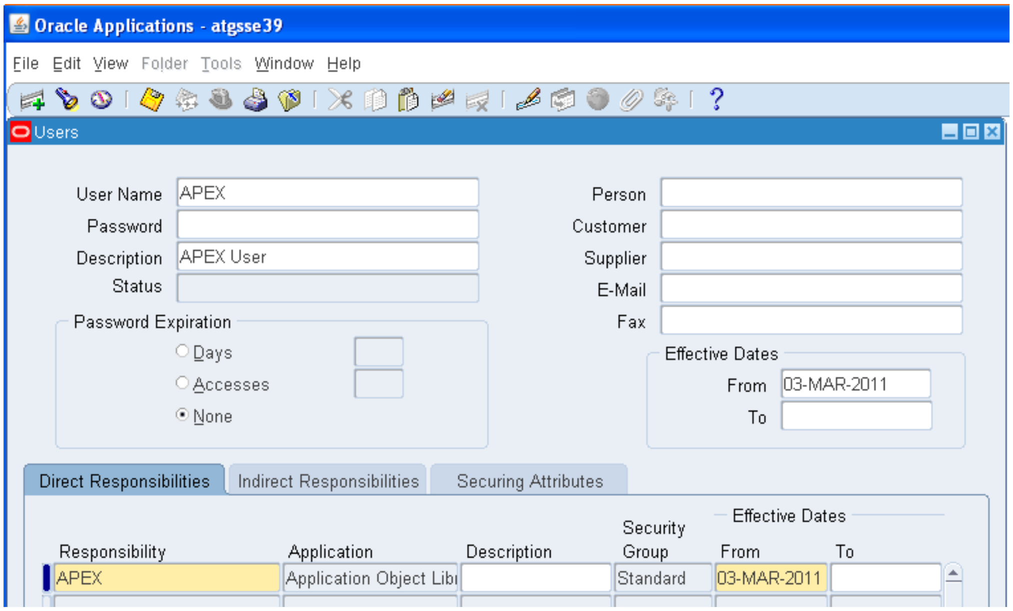
Figure 19. Creating Oracle E-Business Suite Responsibility

Perform the following steps to add a user:

1. Navigate to the System Administrator responsibility> Security > User >Define option
2. Create a new user -

   User Name: **APEX**
   Password: **Welcome0**
   Direct Responsibilities:
      Responsibility: **APEX**
      Application: **Application Object Library**
      Security Group: **Standard**
3. Save the user

4. Navigate to *System Administrator responsibility* > *Concurrent* > *Requests*, click *Find*, check that the **Compile Security Menu** request has completed successfully
5. Navigate to Functional Administrator responsibility> Core Services > Caching Framework, click Global Configuration, click Clear All Cache, click Yes
6. Log out of Oracle E-Business Suite
7. Log in to Oracle E-Business Suite with the APEX user (**http://<EBS_Hostname>:8074/OA_HTML/AppsLogin**)
8. Reset the APEX password **Welcome1**



User Name	APEX
Password	
Description	APEX User
Status	
Person	
Customer	
Supplier	
E-Mail	
Fax	

Password Expiration

○ Days
○ Accesses
● None

Effective Dates

From	03-MAR-2011
To	

				Effective Dates	
Responsibility	Application	Description	Security Group	From	To
APEX	Application Object Libı		Standard	03-MAR-2011	

Figure 20. Creating Oracle E-Business Suite User

Running the Application Express Extension

Now that you have successfully built the Oracle Application Express extension you can run Oracle E-Business Suite and select your new menu links to access your Oracle Application Express application. Follow these steps to test your application:

1. Log in to Oracle E-Business Suite using APEX / Welcome1 (**http://<EBS_Hostname>:8074/OA_HTML/AppsLogin**)

2. Navigate to the APEX Menu

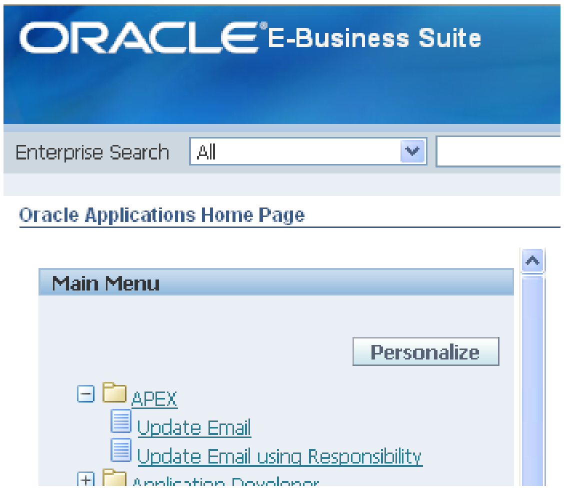
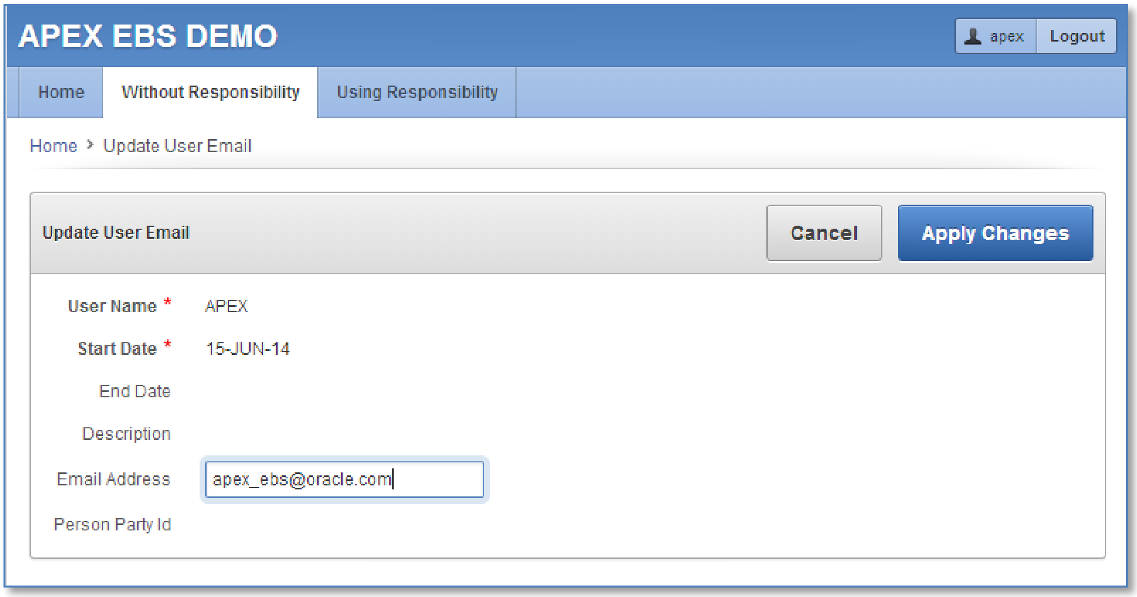


Figure 21. Oracle E-Business Suite runtime menu

3. Select *Update Email* – This will redirect you to the Oracle Application Express login page or to the application depending on if you have configured OSSO above

4. If you need to log into Oracle Application Express - enter *Username* **APEX**, *Password* <<***Welcome1***>>, click *Login*

5. Update the email address, click *Apply Changes* – This will return you to the application home page



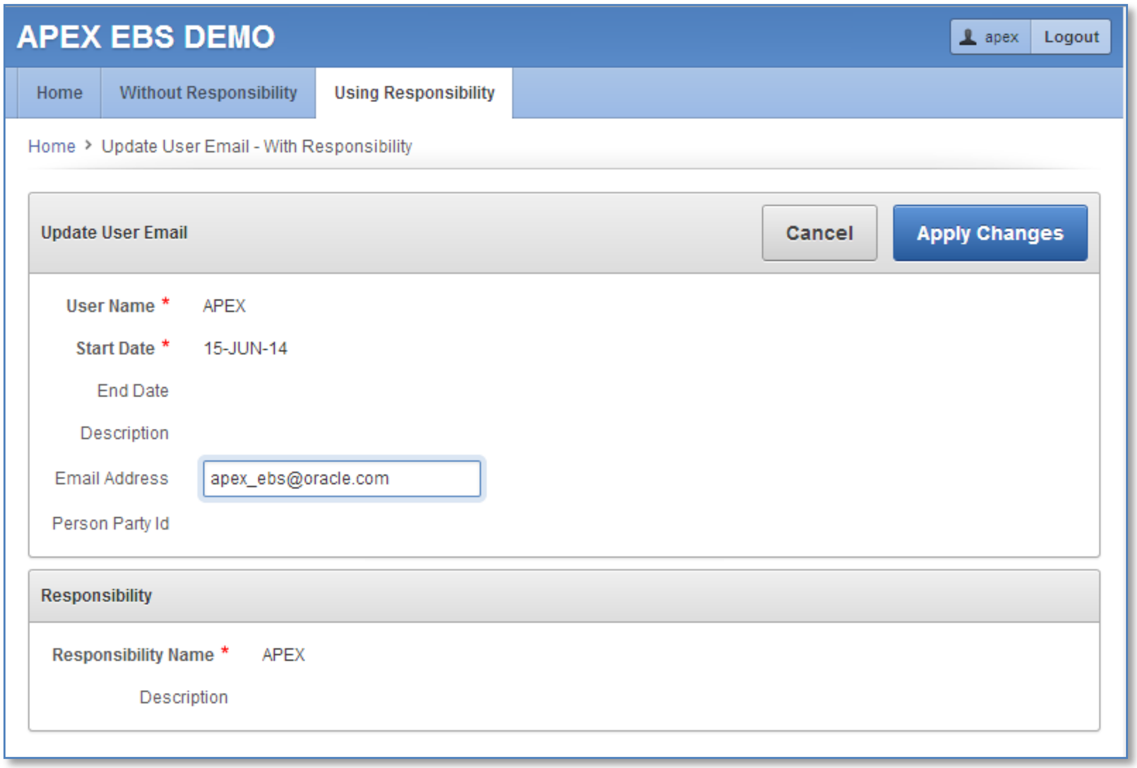
User Name *	APEX
Start Date *	15-JUN-14
End Date	
Description	
Email Address	apex_ebs@oracle.com
Person Party Id	

Figure 22. Oracle Application Express application page for updating user email

6. Click *Using Responsibility* list item or tab – This will produce an authentication error as you navigated to the application without responsibilities

7. Navigate back to Oracle E-Business Suite

8. Select *Update Email using Responsibility*– This will redirect you to the Oracle Application Express application and set your Oracle E-Business Suite responsibility. The responsibility region will display your current responsibility



Update User Email

User Name *	APEX
Start Date *	15-JUN-14
End Date	
Description	
Email Address	apex_ebs@oracle.com
Person Party Id	

Responsibility

Responsibility Name *	APEX
Description	

Figure 23. Oracle Application Express application page showing responsibility

Upgrading your Application Express Extensions

Now that you have your initial Application Express extension to E-Business Suite deployed, it is important to understand how to enhance your Application Express application(s), within this integrated environment.

To fix bugs, modify existing functionality, enhance an application, or implement a new application, you will want to import an Application Express application into your QA / Test and then Production environments. This application import may also require database object upgrades.

When there is no modification or new database objects required, for the enhanced application(s), then the Application Express application(s) can simply be imported into the relevant environments. This can be performed without impacting the Oracle E-Business Suite environment and while users are utilizing the extensions. The user will pick up the latest application definition when they next access the imported application. If they were already in that application, then they will pick up the new application definition when they next request or submit a page.

If an updated or new Application Express application requires only new database objects, such as a new table, view, or database package, then these new database objects should be implemented into the environment before the Application Express application is imported. For Oracle E-Business Suite Release 12.2 environments, Oracle recommends creating these new database objects outside of the E-Business Suite Online Patching Cycle. Once the new database objects are available then the Application Express application can be imported. Again this can be performed while users are utilizing the extensions.

However, if an updated or new Application Express application requires modification or updates to existing database objects or existing data then this should be performed while users are prevented from accessing the extensions. For Oracle E-Business Suite Release 12.1 and prior, this can be achieved by stopping Oracle WebLogic Server, modifying the database objects and data as required, importing the Application Express application, and then restarting Oracle WebLogic Server. For Oracle E-Business Suite Release 12.2 environments, Oracle recommends implementing these changes within an E-Business Suite Online Patching Cycle. The following steps outline how to deploy such applications:

1. Initiate the Oracle E-Business Suite Online Patching Cycle, by running the Online Patching Prepare phase. Review the Oracle E-Business Suite Maintenance Guide R12.2, Chapter 3 “Patching Procedures”, section “The Online Patching Cycle” for further details.
2. Run scripts to modify existing database objects and data within the “patch” edition
3. Run scripts to create new database objects within the “patch” edition (if required)
4. Shutdown the Oracle WebLogic Server used by Oracle Application Express

5. Import updated and new Application Express applications using SQL*Plus
6. Complete the Oracle E-Business Suite Online Patching Cycle
7. Restart the Oracle WebLogic Server used by Oracle Application Express

For example, if you wanted to enhance the application developed earlier in this paper to allow users to update "Known As" as well as their email address then you would need to modify existing database objects before importing the enhanced Application Express application. You would need to update the `APEX_EBS_USER` view to include the `known_as` column from the `FND_USERS` table, and also update the `apex_update_email` function within the `apex_integration_sample_apis` package to accept an extra parameter and update the API call.

In an Oracle E-Business Suite Release 12.2 environment, the Oracle Database feature Edition-Based Redefinition (EBR) is employed within your database for Oracle E-Business Suite Release 12.2 Online Patching. You should utilize this capability to deploy changes to database objects within the schema(s) associated with Oracle Application Express workspaces, such as APEX_EBS_EXTENSION. The Oracle recommendation above allows the Database Administrators to implement all modifications to database objects within the patch edition, so that users are not impacted, except for minimal downtime when the Oracle WebLogic Server is shutdown while the Application Express application is imported. In fact, it is critical to ensure that you do not implement such database object changes in the run edition, as any changes will be overwritten once the E-Business Suite Online Patch Cycle is completed.

Given that the Oracle Application Express Engine (For Application Express Release 4.2 that schema is APEX_040200) is not editioned, when you import a new Application Express application definition, it will be "live" immediately to all users. This is irrespective of if Oracle E-Business Suite is currently in Online Patching Cycle or not. If the new application relies on updated database objects and those updates have been performed in the patch edition, then your applications will not see those updated database objects as the will be referencing the run edition. Therefore, it is very important to coordinate when required updates to underlying database objects and when Application Express applications are performed, to minimize disruption to users.

Upgrading your Application Express Extensions in a Different Database

When you have installed Oracle Application Express into a different Oracle Database to the database where E-Business Suite is installed, then existing database objects may need to be modified in the local database, where Oracle Application Express is installed, or in the Oracle E-Business Suite database, or in both databases.

If an updated or new Application Express application requires database objects within the local database (within the APEX_LOCAL schema) to be modified then you should prevent user access while the database objects are modified and the application(s) are being imported. This can be achieved by stopping Oracle WebLogic Server, modifying the database objects and data as required, importing the Application Express application, and then restarting Oracle WebLogic Server.

If an updated or new Application Express application requires only new database objects within the Oracle E-Business Suite database (within either the APPS or APEX_EBS_EXTENSION schemas), such as a new table, view, or database package, then these new database objects should be implemented into the environment before the Application Express application is imported. For Oracle E-Business Suite Release 12.2 environments, Oracle recommends creating these new database objects outside of the E-Business Suite Online Patching Cycle. Once the new database objects are available then the Application Express application can be imported.

However, if an updated or new Application Express application requires modification or updates to existing database objects or existing data, within the Oracle E-Business Suite database, for Oracle E-Business Suite Release 12.2 environments, Oracle recommends implementing these changes within an E-Business Suite Online Patching Cycle. The following steps outline how to deploy such applications:

1. Initiate the Oracle E-Business Suite Online Patching Cycle, by running the Online Patching Prepare phase. Review the Oracle E-Business Suite Maintenance Guide R12.2, Chapter 3 “Patching Procedures”, section “The Online Patching Cycle” for further details.
2. Run scripts to modify existing database objects and data within the “patch” edition
3. Run scripts to create new database objects within the “patch” edition (if required)
4. Shutdown the Oracle WebLogic Server used by Oracle Application Express
5. Run scripts to modify existing database objects and data on the local database
6. Import updated and new Application Express applications using SQL*Plus
7. Complete the Oracle E-Business Suite Online Patching Cycle
8. Restart the Oracle WebLogic Server used by Oracle Application Express

Conclusion

Oracle Application Express is an excellent complement to Oracle E-Business Suite. As a development framework, Oracle Application Express allows cross-functional teams to easily collaborate and quickly deliver solid custom extensions to Oracle E-Business Suite. The Oracle Application Express footprint on an Oracle E-Business Suite environment is negligible, but allows deeply-integrated extensions using only standard customizations within the Oracle E-Business Suite implementation.

Use of a separate Oracle Application Express schema with only the minimum required privileges for Oracle E-Business Suite objects ensures that Oracle Application Express applications only have access to the interfaces and data required for their operation. This minimizes the security risk to the Oracle E-Business Suite inherent in making any extensions. By integrating with Oracle E-Business Suite Function Security (Authorization), Oracle Application Express extensions utilize a single point of security administration. When a responsibility is added or removed from a user within Oracle E-Business Suite, the user's access to the associated functionality in Oracle Application Express applications is affected likewise.

Further, the advantage of utilizing Oracle Application Express over building extensions from scratch is that Oracle Application Express provides a hardened declarative framework to minimize security vulnerabilities. Oracle Application Express release procedures include extensive security testing to minimize the threat of security breaches from cross site scripting and SQL injection.

Oracle Application Express gives you the power to quickly create forms, reports and complete applications rapidly, with little or no programming. With the proper deployment architecture, as outlined in this paper, you can use Oracle Application Express to extend the capabilities of your Oracle E-Business Suite, using the same data for both sets of applications, sharing user authentication, and calling applications and components seamlessly.

Acknowledgements

This paper is a collaboration between the Oracle Application Express and Oracle E-Business Suite teams. George Buzsaki, Oracle E-Business Suite Architecture, has provided significant input to ensure the solutions provided meets the Oracle E-Business Suite best practices and presents fully supported configurations.

Content assistance was provided by Rod West from Cabot Consulting (rodwest@cabotconsulting.co.uk) and Wolfgang Moritz from Farwest Steel (wolfgang.moritz@farweststeel.com).

Oracle is committed to developing practices and products that help protect the environment

Extending Oracle E-Business Suite
Release 12.1 and above using
Oracle Application Express
Revision 2

September 2014
Author: David Peake
Contributing Authors: Kevin Hudson,
Joel Kallman, Santiago Bastidas

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Hardware and Software, Engineered to Work Together